SecureNoC: Enhancing On-Chip Network Security for Many Integrated Core Systems

Xiaoming Guo, Sijia He, Amlan Nayak, Jay Zhang

EECS 578: Correct Operation for Processors & Embedded Systems, Fall 2015

UNIVERSITY OF MICHIGAN

MICHIGAN ENGINEERING
UNIVERSITY OF MICHIGAN

Introduction

Motivation

- Networks-on-Chip (NoCs) have emerged as the most promising and readily implementable interconnect solution for Chip Multiprocessors and SoCs
- NoCs face security vulnerabilities that are similar to existing macro-scale networks

Vulnerabilities Addressed

- Denial-of-Service (DoS) attack
  - One or more nodes in an NoC inject packets in an abnormal high rate, causing delay or failure of other packets' transmission
- Extraction of Secure Information
  - A compromised core is able to access secured packets via local port during transmission

Contribution

- A high-packet injection detector with false positive prevention to handle multiple-node DoS attack
- Secure Packet ExchAngER (SPEAR) system which enables a method for a pair of nodes in an NoC to send and receive secure packets by eliminating the possibility of intermediate nodes from snooping their contents.

Implementation

Figure 1: DoS detection phase 1

- Count the number of flits injected from the local port per epoch
- Temporarily stop accepting packets for two epochs if count number exceeds the threshold
- Notify the core to reschedule its packet injection process

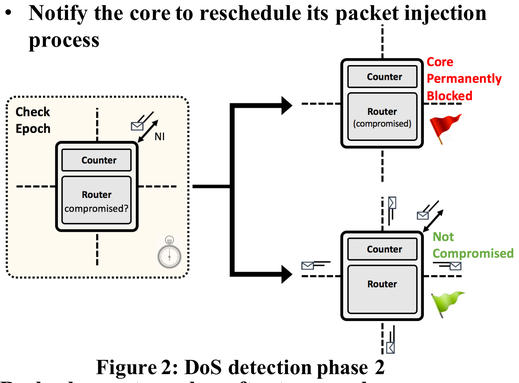

Figure 2: DoS detection phase 2

- Recheck count number after two epochs
  - Permanently block the core if the count number exceeds the threshold again
  - Resume normal operation if the count number falls below the threshold

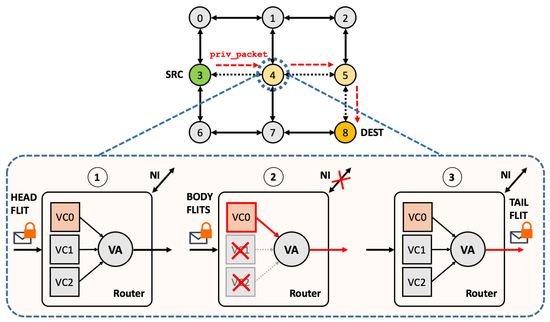

Figure 3: SPEAR operational phases

- Support secure exchange of privileged packets
- Reserve Virtual Channel 0 (VC 0) for privileged packets only
- When head flit of a privileged packet arrives:
  - Block other VCs
  - Cut off connection to local port
- When tail flit of a privileged packet arrives:
  - Unblock other VCs
  - Re-enable connection to local port

Experimental Setup

The two security mechanisms were implemented both in simulation and hardware.

- Booksim, a cycle-accurate NoC simulator was used to test the design functionality and to obtain high-level performance results such as packet latency
- The designs were then ported to Verilog in order to determine area and delay penalties.

Results

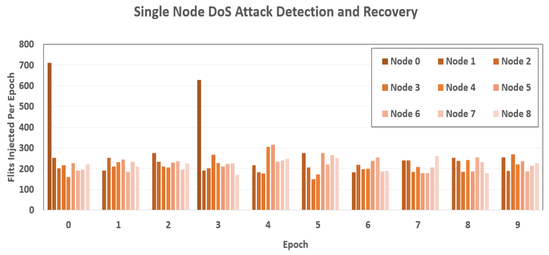

Figure 4: DoS attack demonstration from node 0

- Single Node DoS attack detection and recovery procedure using traffic monitoring unit:
  - Node 0 exceeds the threshold at epoch 0
  - Temporally block node 0 and allow rescheduling during epoch 1 and epoch 2
  - Node 0 exceeds the threshold again at epoch 3
  - Permanently block node 0

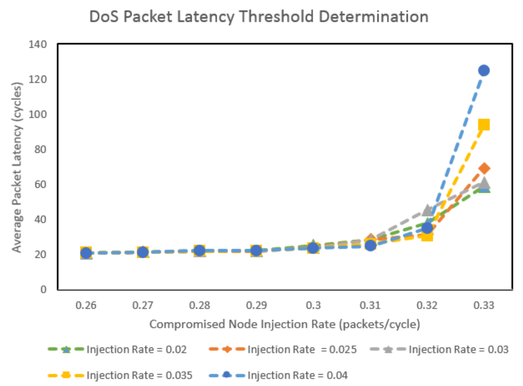

Figure 5: DoS threshold experiment

- DoS injection rate threshold shows little dependency on the injection rate of other nodes

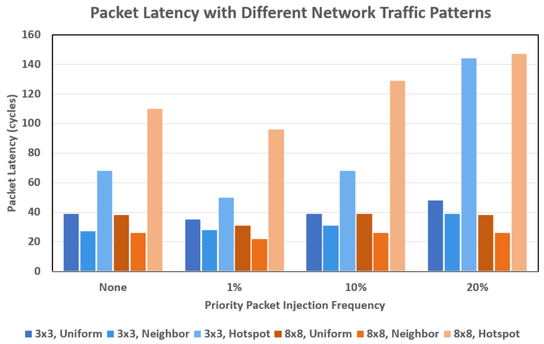

Figure 6: Latency tests with traffic patterns

- Significant performance impact can be only observed with Hotspot traffic in 3x3 mesh network

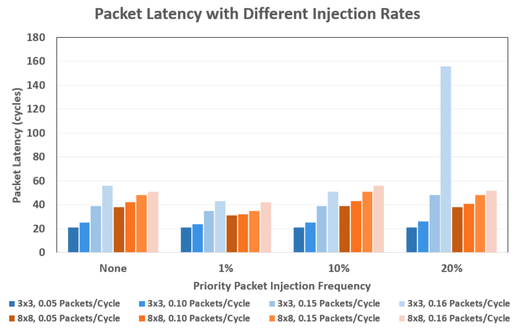

Figure 7: Latency tests with injection rates

- For 3x3 mesh, the performance impact is only noticeable when injection rate reaches 0.16 packets/cycle with 20% privileged packet frequency
- For 8x8 mesh, the performance impact is negligible for all tests

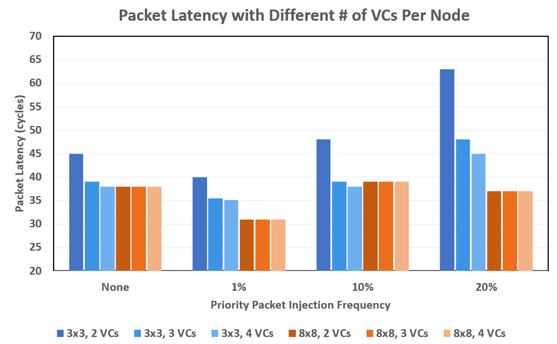

Figure 8: Latency tests with different # of VCs

- For 3x3 mesh, the performance impact of having a dedicated VC is not significant when number of VCs >=3
- For 8x8 mesh, the performance impact is negligible for all tests

Hardware Overhead

	Clock Speed Slowdown	Area Overhead
DoS	0	0.66%
DoS + SPEAR	0	1.1%

Table 1: Hardware overhead analysis

Future Work

In order to improve VC allocation performance, instead of starving non-privileged VCs during secure packet exchange, the SPEAR mechanism can be augmented with a time-division multiplexing scheme. The credit-return policy of the router can be modified to achieve a finer grain control over the local core's packet injection rate. This would improve the efficiency of the DoS recovery mechanism.

Conclusion

As the complexity of NoC designs increases, security vulnerabilities arise and need to be addressed. Through large-scale network security is a well-studied topic, little attention has been paid to on-chip networks. In this work, we designed and evaluated mechanisms to protect NoCs from two major forms of attacks, namely DoS and Side-Channel information extraction. Further, we demonstrated that these mechanisms can be readily implemented with minimal changes to existing NoC infrastructures and with negligible performance and area overheads.