Camera Integration for hardware-in-the-loop simulator avionics demonstrator

Chaitra Prasad Niddodi
chaitra@illinois.edu

Chien-Ying Chen
cchen140@illinois.edu

Sibin Mohan
sibin@illinois.edu

University of Illinois at Urbana-Champaign, United States

1. INTRODUCTION

The increasing number of attacks on real time systems has made it necessary to re-design these systems to include security as an integral part of the resource management algorithms. Here we discuss the implementation of a hardware-in-the-loop simulator (Avionics Demonstrator) to demonstrate the integration of security into its design.

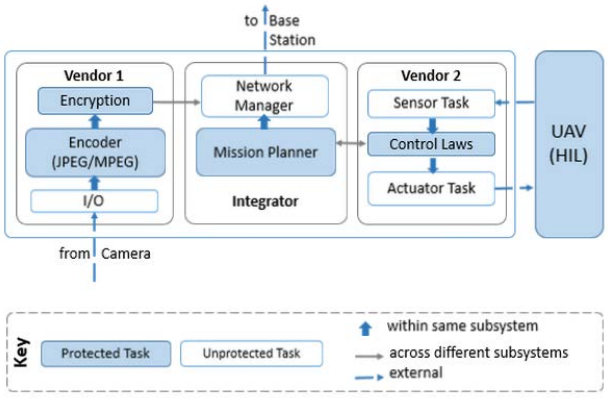

Fig 1. Avionics Demonstrator

2. MOTIVATION

Real time systems are being extensively used in monitoring a variety of systems across many domains. Their designs are getting increasingly complex. Real time systems are also being interconnected, often in an insecure manner over the internet. All these make them more vulnerable to malicious attacks. Hence security in real time systems has become very essential.

3. CHALLENGES

- The constrained environment in real time systems with respect to processing power and memory capacity.
- Tasks are also being constrained with respect to time.
- Synchronization of hardware (FPGA) and software (processor) to work together.
- Debugging applications developed on real time systems requires debugging of hardware and software separately.

4. SYSTEM IMPLEMENTATION

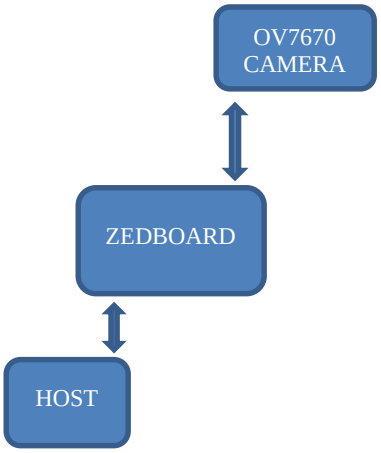

Fig 2. Schematic of system implemented

The defining feature of Zedbaord is that it combines a dual-core ARM Cortex-A9 processor with traditional Field Programmable Gate Array (FPGA) logic fabric.

Server running on host

Python 3 UDP server performs AES decryption on the encrypted video obtained from the OV7670 camera through the UDP client socket on zedboard and displays it.

FreeRTOS tasks running on zedboard

The Zynq processor on the zedboard runs FreeRTOS. There are two tasks running on FreeRTOS.

1. AES Encryption - This task performs AES encryption for the video obtained from OV7670 camera. The key used for encryption is 128 bit long. The mode of AES encryption performed is ECB (electronic code book).
2. UDP client socket - This task creates a UDP client socket. lwIP socket API is used. Video encrypted by AES task is sent to the server running on the host.

Piping OV7670 video to host server

OV7670 camera driver is implemented on the PL (programmable logic) on zedboard (FPGA). The 640 x 480, 30 frames per second video captured by the camera is stored in a Block RAM based frame buffer. This video is then encrypted by the AES task running on FREERTOS. Finally, the encrypted video is sent to the host server by the UDP client task running on FREERTOS.

5. CONCLUDING REMARKS

We are currently working on the driver for the OV7670 camera on zedboard. The driver is implemented on the FPGA. We are first testing the driver by piping the camera video to the VGA output on zedboard.

6. REFERENCES

1. http://www.freertos.org
2. https://pypi.python.org/pypi/pycrypto
3. http://bradconte.com/aes_c
4. http://lauri.xn--vsandi-pxa.com/hdl/zynq/zybo-ov7670-to-vga.html
5. http://hamsterworks.co.nz/mediawiki/index.php/OV7670_camera