Background

In a time where every electronic device is connected to the Internet, security in embedded systems is becoming more and more important. Isolation is a fundamental principle in security, which in embedded systems is provided by a separation mechanism such as a traditional operative system or a tiny RTOS. But in the context of security applications, to ease evaluation and to reduce the risk for software bugs, it is normally preferred that the separation software is kept as simple as possible. This has led to the introduction of thin hypervisors and separation kernels in security sensitive applications.

During 2014, Business Security AB in cooperation with the Swedish Institute of Computer Science (SICS) in Lund is offering two master theses in the area of embedded system security. The first thesis is about design, implementation and evaluation of a separation kernel / thin hypervisor while the second thesis is about performance improvements within one such environment.

Objectives for thesis 1: design and implementation

Consider the simple embedded system illustrated in Figure 1. Here, three different software components are executing side by side. Furthermore, a separation mechanism is used to ensure isolation between these components. In this example, notice that the separation mechanism prohibits components from directly accessing hardware resources (hardware in this context is the CPU and its peripherals) or manipulating other components.

This is very similar to virtualization on desktop computers (which allows you to run multiple operative systems on a single machine), but in this thesis we are interested in a mechanism that is much simpler to implement (i.e. around 2,000 lines of code) and verify. The software components utilizing this mechanism may also be much simpler than a full-fledged operative system.

Your task in this thesis is to design and implement a minimal separation kernel:

1. Given a security case and a hardware target, analyse the security requirements and the target environment.
2. Implement a simple separation kernel, either from scratch or by extending the SICS Thin Hypervisor (bitbucket.org/sicssec/sth).
3. Depending on your interests, perform one of the following:
a) Demonstrate the security of the implemented solution. This will require extensive testing and documentation.
b) Extend the implemented solution to be able to host instances of full-scale operative systems such as Linux.

4. Provide a written report (i.e. your thesis).

Objectives for thesis 2: high performance and secure communication

Even isolated software components sometimes need to communicate. In a secure environment, while direct access is prohibited, an indirect and supervised communication channel must be provided.

Consider for example the embedded system in Figure 2, where components 1 and 5 are each given access to an external interface while components 2 and 3 have an indirect communication channel. The separation kernel in this case will enforce the communication policies and ensure that no security functions are bypassed.

Your task in this thesis is to extend the separation kernel with high performance (high throughput, low latency) secure communication channels:

1. Given a security case and a hardware target, analyse the security requirements and the target environment.
2. Extend an existing separation kernel with high performance communication channels.
3. Demonstrate the performance of the implemented solution. This will require extensive testing under different loads and data characteristics.
4. Provide a written report (i.e. your thesis).

Requirements

We are looking for students who meet the following requirements:

- Knowledge in C or assembly (advanced knowledge is a big plus),
- Basic knowledge in modern CPU architecture and/or operating system architecture,
- Good spoken and written English and Swedish.

Furthermore, the following qualities are desired but not required:

- Experience in low-level programming, preferably on ARM or MIPS,
- Experience in kernel development, preferably with Linux or OpenBSD,
- Experience in open source development and use of Git or Subversion.
How to apply

Applications should include a brief personal letter, your CV with your education, professional experience and specific skills and recent grades. In your application, make sure to give examples of previous programming or other projects that you consider relevant for the position. Candidates are encouraged to send in their application as soon as possible, in paper form or via e-mail. Suitable applicants will be interviewed as applications are received.

The contact for both theses at SICS is Arash Vahidi. Please email your application to arash@sics.se or Arash Vahidi +46 (0)70-7731545
SICS Swedish ICT AB
Ideon Science Park - b2
Scheelevägen 17
SE-223 70 Lund
Sweden

Don't forget to specify which of the two theses you are interested in!

Contacts for the theses at LTH are Ben Smeets and Martin Hell:

Ben Smeets T: +46 (0)46 2227521, E: ben.smeets@eit.lth.se
Martin Hell T: +46 (0)46 2224353, E: martin.hell@eit.lth.se

About Business Security

One of Europe’s leading companies within cryptology and hardware based cryptographic systems. Our VPN system SecuriVPN is used by customers with demanding requirements on security and reliability. For over 20 years we have delivered cryptographic systems and security solutions to companies and organizations in over 40 countries world wide.

About SICS

The Swedish Institute of Computer Science (SICS) is a leading research institute for applied information and communication technology in Sweden. We are a non-profit-distributing organization with main offices in Kista outside Stockholm and smaller offices in Uppsala and Lund. SICS employs approx. 140 researchers, including 45 PhDs, and hosts another 30 researchers from KTH, consultants and students working on their Master Thesis.