Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pervasive and Mobile Computing 1 (2005) 425–445
pervasive and mobile computing
www.elsevier.com/locate/pmc

Sizzle: A standards-based end-to-end security architecture for the embedded Internet☆

Vipul Gupta*, Michael Wurm, Yu Zhu, Matthew Millard, Stephen Fung, Nils Gura, Hans Eberle, Sheueling Chang Shantz

Sun Microsystems Laboratories, 16 Network Circle, UMPK16 160, Menlo Park, CA 94025, USA

Received 4 June 2005; accepted 22 August 2005
Available online 10 October 2005

Abstract

According to popular perception, public-key cryptography is beyond the capabilities of highly constrained, "mote"-like, embedded devices. We show that elliptic curve cryptography not only makes public-key cryptography feasible on these devices, it allows one to create a complete secure web server stack that runs efficiently within very tight resource constraints. Our small-footprint HTTPS stack, nicknamed Sizzle, has been implemented on multiple generations of the Berkeley/Crossbow motes where it runs in less than 4 KB of RAM, completes a full SSL handshake in 1 s (session reuse takes 0.5 s) and transfers 1 KB of application data over SSL in 0.4 s. Sizzle is the world's smallest secure web server and can be embedded inside home appliances, personal medical devices, etc., allowing them to be monitored and controlled remotely via a web browser without sacrificing end-to-end security.

Keywords: Sensor network security; Elliptic Curve Cryptography; Smallest secure webserver

☆ Expanded version of a paper that received the Mark Weiser Best Paper Award at PerCom 2005.

* Corresponding author. Tel.: +1 650 786 7551; fax: +1 650 786 6013.
E-mail addresses: vipul.gupta@sun.com (V. Gupta), mwurm@sime.com (M. Wurm), davidyuzhu@hotmail.com (Y. Zhu), mmillard@kos.net (M. Millard), stephen.fung@gmail.com (S. Fung), nils.gura@sun.com (N. Gura), hans.eberle@sun.com (H. Eberle), sheueling.chang@sun.com (S. Chang Shantz).

1574-1192/$ - see front matter
doi:10.1016/j.pmcj.2005.08.005

1. Introduction

In the last several years, the Internet has grown rapidly beyond servers, desktops and laptops to include handheld devices like PDAs and smart phones. There is now a growing realization that this trend will continue as increasing numbers of even simpler, more constrained devices (sensors, home appliances, personal medical devices) get connected to the Internet. The term "embedded Internet" is often used to refer to the part of the Internet that is invisibly and tightly woven into our daily lives. Embedded devices with sensing and communication capabilities will enable the application of computing technologies in settings where they are unusual today: habitat monitoring [25], medical emergency response [31], battlefield management and home automation.

Many of these applications have security requirements. For example, health information must only be made available to authorized personnel (authentication) and be protected from modification (data integrity) or disclosure (confidentiality) in transit. Even seemingly innocuous data such as temperature and pressure readings may need to be secured. Consider the case of a chemical plant where sensors are used to continuously monitor the reactions used in manufacturing the final product. Without adequate security, an attacker could feed highly abnormal readings into the monitoring system and trigger catastrophic reactions.

Secure Sockets Layer (SSL)[1] [10] is the most commonly used security protocol on the Internet today. It is built into many popular applications, including all well-known web browsers, and is widely trusted to secure sensitive transactions including on-line banking, stock trading and e-commerce. This paper describes our investigation into using the same protocol to secure the embedded Internet.

SSL combines public-key cryptography for key-distribution and authentication with symmetric-key cryptography for data encryption and integrity. Public-key cryptography is widely believed to be beyond the capabilities of embedded devices. This perception is primarily driven by earlier experiments involving C-language implementations of RSA, today's dominant public-key cryptosystem [30].

Recent work in our research group has shown that optimized, assembly-language implementations of RSA perform much better and the use of Elliptic Curve Cryptography (ECC) provides an additional order of magnitude performance improvement on 8-bit CPUs [15]. First proposed by Victor Miller [20] in 1985, and independently by Neal Koblitz [18], ECC is emerging as an attractive alternative to RSA for resource-constrained environments.

We have built a small-footprint secure web server stack (including HTTP and SSL), called Sizzle[2] on top of our ECC and RSA implementations. Sizzle runs efficiently under tight resource constraints and interoperates with browsers like Mozilla, Internet Explorer and Safari. It can take advantage of the higher performance of ECC when communicating with an ECC-enabled version of Mozilla [11].

The main contributions of this paper are:

[1] Throughout this paper, we use SSL to refer to all versions of this protocol including version 3.1 *aka* Transport Layer Security (TLSv1.0) [7].

[2] This name derives from "Slim SSL" (SSSL).