Keywords: embedded security, secure authenticator, SHA-256, ECDSA computing

APPLICATION NOTE 5931

DEVICE AUTHENTICATION THWARTS COUNTERFEITING

By: Michael D'Onofrio

Abstract: An embedded secure authenticator protects end-users and OEMs from counterfeit devices. Secure authentication verifies to a host system that an attached device is genuine and can be trusted.

A similar version of this article appeared in the July 28, 2014 edition of *Pulse Magazine*.

Introduction

A primary method of providing electronic security is through the use of a secure authentication scheme in hardware. Device authentication is used to protect end users and OEMs from counterfeit peripherals, sensors, consumables, or other devices. It is a method that verifies to the host system that an attached device is genuine and can be trusted.

The Problem of Counterfeit Devices

Let us start by agreeing on what "counterfeit" means in our discussion. A counterfeit device could simply be a cheap clone of the original. Consider, for instance, a medical sensor that plugs into a control module. It is carefully manufactured to look and act the same, but the device's quality and the accuracy of its data will be questionable, possibly leading to a misdiagnosis and incorrect treatment.[1] Clearly with a cloned counterfeit instrument like this, device authentication would protect patients from faulty equipment and healthcare providers from the liability of flawed professional care.[2]

A less dire and perhaps more commonplace example of counterfeiting involves inkjet printer cartridges. Many OEMs sell their printer at or near cost with the expectation of generating profits and recovering R&D costs from the sale of the disposable inkjet cartridges. But when cloned inkjet cartridges are packaged and sold as genuine articles, the OEM is cheated and loses revenue. Counterfeit, poor-quality cartridges can also fail and damage the printer, which then hurts the brand reputation of the OEM.

As the world and our many electronic devices become more interconnected, device manufacturers have become increasingly aware of the security threats and the potential damaging impact of counterfeit devices. It is thus no surprise that OEMs are implementing various levels of security to detect and thwart the counterfeits.

Authentication Methods

Electronic security today encompasses a range of security methods including cryptographic algorithms and authentication protocols, some stronger than others depending on the application.

A simple authentication method works much like an ID. As long as the host "master" system receives the correct ID data from a peripheral "slave" device, that slave is assumed to be authentic. The problem with this method is that the ID data itself is exposed during communication from slave to host, and is then accessible to a hacker. This scheme is easily bypassed by recording or replaying the ID data and then acting as an authentic device.

Another method shown to be very robust is the use of a one-way hash function that is easy to generate, but nearly impossible for a hacker to invert to discover the input elements. SHA-256 is a well-tested and proven hash function. Using a challenge-and-response protocol, SHA-256 functions calculate a message authentication code (MAC) based on multiple public and private data elements. A peripheral authentication IC calculates a MAC and sends it to the host system; the host calculates its own MAC, presumably with the same input. Then the host compares its own MAC to that of the peripheral and, if they match, the peripheral's authenticity is ensured. As long as the private data elements remain secure, SHA-256 authentication provides a very high level of certainty that a peripheral's MAC is indeed authentic.[3, 4]

SHA-256 algorithms can be implemented in software on both host and peripheral devices, but software implementations can be tricky to implement. A hacker who can break the controller protections and decompile the code will have access to the secrets used in the authentication process. Consequently, a hardware-based SHA-256 secure authentication system using a secure authentication IC like a DeepCover® DS28E15 is stronger.[5]

A DeepCover secure authenticator[6] provides the benefits of a one-way hash function and the security of hardware-based cryptography. For SHA-256 implementations, the coprocessor stores data elements, such as the host-side secret, securely in protected memory that can be used as an input to the algorithm, but not read out. The coprocessor also performs the SHA-256 computations for the host side. To perform this verification, the MCU sends a random challenge to the coprocessor and authenticator, and the MCU collects the MAC responses from the coprocessor and authenticator for comparison. Countermeasures are implemented on the hardware authentication ICs to make it nearly impossible to extract the sensitive data or create a workaround by physical examination of the device.

To avoid the need to protect a host-side secret, public key-based algorithms like ECDSA[7] do not use a host-side secret. Rather, a private key is securely stored in a secure authentication IC, and a public key is used by the host to verify the authenticity of the peripheral IC. For ECDSA implementations, the coprocessor does not need to protect the host-side public key, but it does perform the computation-intensive ECDSA computing and reports the result back to the MCU.

A hardware-based approach provides a more attack-resistant secure authentication platform, and also reduces the time and cost spent on development within the host and peripherals.[8]

Implementation of a DeepCover secure authenticator is simple using a 1-Wire® interface to communicate between coprocessor and authenticator. The coprocessors have an integrated I^2C to 1-Wire bridge. The 1-Wire authenticators are powered parasitically over the I/O pin to further simplify integration on the peripheral side. The 1-Wire interface supports multidrop, so that multiple 1-Wire authenticators can be used on the same I/O bus if the application requires it.

Summary

Secure authentication of peripheral devices is critical for protecting customers and OEMs from counterfeit devices. Many authentication methods exist, each with its own strengths and vulnerabilities. Hardware-based secure authentication using a DeepCover secure authenticator provides the right balance between security, bill-of-materials (BOM) cost, and ease of integration into new and existing designs.

References

1. For a discussion of how potential counterfeit medical devices can impact health care see Tremlet, Christophe, “Hardware Security ICs Offer Large Security Returns at a Low Cost,” **Electronic Products**, 08/20/13, www.electronicproducts.com/Software/EDA_Software_and_Hardware/Hardware_Security_ICs_Offer_Large_Security_Returns_at_a_Low_Cost.aspx#.U7wlp0A-eSo; also available as Maxim Integrated application note 5716.
2. Read how counterfeit utility meters resulted in major financial losses to a Puerto Pico utility and how embedded security could have prevented the problem. See Andeen, David, “Smart-grid security: history demonstrates need,” **Power Systems Design Europe**, December 2012, pp. 33-36, www.powersystemsdesign.com/library/resources/images/IssuePDF/psde_december12.pdf; also available as Maxim Integrated application note 5537.
   For a more general discussion of how counterfeit devices can impact the smart grid and industrial control, see also Andeen, David, “Energy Measurement and Security for the Smart Grid - Too Long Overlooked,” at **EETimes China**, January 7, 2013, http://archive.eet-china.com/www.eet-china.com/ART_8800680222_617693_TA_a2c42dcf.HTM (in Chinese), and Maxim Integrated application note 5536.
3. For a detailed discussion of a SHA-256 master/slave authentication system, see Linke, Bernhard, “The Fundamentals of a SHA-256 Master/Slave Authentication System,” **EETIMES**, June 19, 2013, www.eetimes.com/document.asp?doc_id=1280942. More detailed information is also available to qualified customers in Maxim Integrated application note 5546.
   Also Linke, Bernhard, “A SHA-256 master/slave authentication system,” **Electronic Products**, 06/16/2014, www.electronicproducts.com/Digital_ICs/Communications_Interface/A_SHA-256_master/slave_authentication_system.aspx#.U7ws6kA-eSo; more details available to qualified readers in Maxim Integrated application note 5785, “Implement Heightened Security with a SHA-256 Master/Slave Authentication System.”
4. Cf. Note 3.
5. For a discussion of how ICs with security embedded in the hardware are the “ultimate protection devices,” see Tremlet, Christophe, “Industrial control systems need security ICs,” **EETimes**, November 7, 2012, www.eetimes.com/document.asp?doc_id=1280102; also at Maxim Integrated application note 5522.
6. Tremlet, Christophe, “How to protect embedded software against attacks,” **New Electronics**, 11 February 2014, www.newelectronics.co.uk/electronics-technology/how-to-protect-embedded-software-against-attacks/59422/; also available as Maxim Integrated application note 5696. See also Note 3.
7. See also Linke, Bernhard, “Using the Elliptic Curve Digital Signature Algorithm effectively,” **Embedded.com**, February 02, 2014, www.embedded.com/design/safety-and-security/4427811/Using-the-Elliptic-Curve-Digital-Signature-Algorithm-effectively; also available as Maxim Integrated application note 5767.
8. For a discussion of how “cradle-to-grave” security measures embedded in hardware can protect industrial and smart grid devices, see Ardis, Kris, “Stuxnet and other things that go bump in the night,” **EDN**, October 11, 2012, www.edn.com/design/systems-design/4398386/2/Stuxnet-and-other-things-that-go-bump-in-the-night; also available as a Maxim Integrated application note 5445. This part series on security for the IoT is also available as Maxim Integrated application note 5725.

Related Parts		
DS28E15	DeepCover Secure Authenticator with 1-Wire SHA-256 and 512-Bit User EEPROM	Free Samples

More Information
For Technical Support: http://www.maximintegrated.com/en/support
For Samples: http://www.maximintegrated.com/en/samples
Other Questions and Comments: http://www.maximintegrated.com/en/contact

Application Note 5931: http://www.maximintegrated.com/en/an5931
APPLICATION NOTE 5931, AN5931, AN 5931, APP5931, Appnote5931, Appnote 5931