![maxim integrated logo]

Keywords: challenge-response, challenge, random challenge, response, SHA, SHA-1, MAC, coprocessor, user, access control, authentication, ecash, web access, random number generator, DS1963S, DS1961S, DS2432, software piracy

APPLICATION NOTE 190

Challenge and Response with 1-Wire® SHA Devices

Mar 12, 2002

Abstract: Challenge-response can be a secure way of protecting access to any privileged material if implemented correctly. In this document, many options for challenge-response access control are discussed but the most secure method given is presenting a different random challenge on each access attempt and having a response that only the host can interpret without giving out any secrets. This document shows why Maxim's SHA-1 iButtons® and 1-Wire devices are ideal choices when implementing this kind of challenge-response system.

Introduction

Challenge-and-response is a common authentication technique whereby some private information is verified due to a response from a given challenge. Most secure systems rely on some form of challenge-response. A simple example of a challenge-response scenario is "What is the password?" The person must supply the correct password or "response" to this challenge. The problem with that challenge-response method is someone could over hear the private information. So more advanced methods are needed. One such method is the ID card, which has specific information to pass. This method can be defeated by duplicating the information on the ID card. Another method supplies a password with the ID card. However if the person has to enter in a password then someone can observe them type it in.

The most secure method would have the challenger send a different challenge every time, preferably a random challenge. The random challenge is taken by the remote device and the response is computed. The host takes the random challenge and computes the correct response. The host response is compared to the response received. If both match then the remote device is valid. This form of challenge-response is more secure because the private information is never revealed and the response is different with each random challenge. See **Figure 1** for a simple diagram of challenge and response.
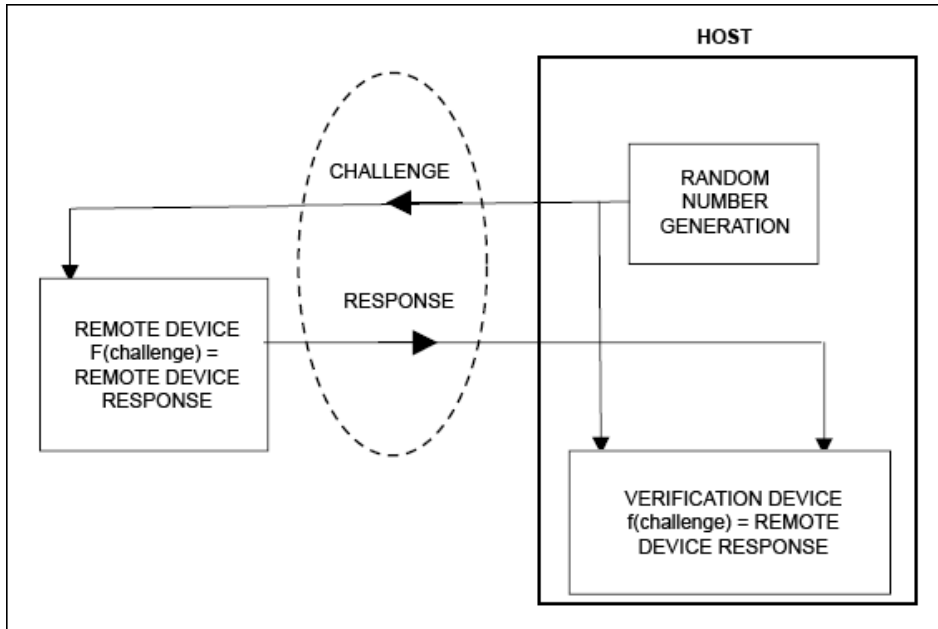
Figure 1. Challenge-Response

The last method describes how the Maxim 1-Wire SHA devices' challenge-response algorithm functions. The host sends a challenge to the device. The device takes the challenge and computes its response. It sends back the response and unique information, which the host then verifies.

There are many applications where the challenge and response method could be used effectively:
1. User authentication
2. Software authorization (binding the software to a hardware token)
3. Web access control
4. eCash
5. Physical access control

Design Specification with SHA Devices

Definitions

A *coprocessor* is used by the host. It creates a random challenge that is sent to the remote device for verification. The coprocessor computes the valid response (the response that it should receive) to verify the device.

A *user* is the remote device. It takes the random challenge and computes the response.

Setup

The coprocessor is a DS1963S initialized for verifying a user token as a member of the system. Initializing the coprocessor consists of two steps:
- Installing the system authentication secret, which is a secret installed for authentication user devices.
- Writing the system configuration data to the device.

The user token is a DS1963S, DS1961S, or a DS2432 that identifies a user to the system. Initializing the user token only consists of one step:
- Installing the master authentication secret and binding it to the device so a unique secret is created for the user.

Challenge and Response Using SHA Devices

To authenticate a user SHA device, the host would ask the coprocessor to compute a random 3-byte challenge and send it to the user device. It also asks the coprocessor to compute a response, which is a 20-byte SHA-1 MAC. The user device takes the challenge, the account data, its own serial number, and its authentication secret to compute the response. The response is then read by the host for comparison to its own computation later. To verify the response, the host asks the coprocessor to first recompute the user's unique device authentication secret using the user device's serial number, and the system

authentication secret. The host then asks the coprocessor to compute a response using the recomputed device authentication secret and the challenge code that it sent over to the user. This coprocessor-computed response is then compared with the response read from the user to determine whether the user device is valid. This two-step process is necessary because the host can't read the user device's secret and because the user device authentication secret is different from the system authentication secret. Note that device authentication only requires the user device to carry the right authentication secret, it does not check the account data contents. **Figure 2** shows this transfer challenge and response method for SHA devices.
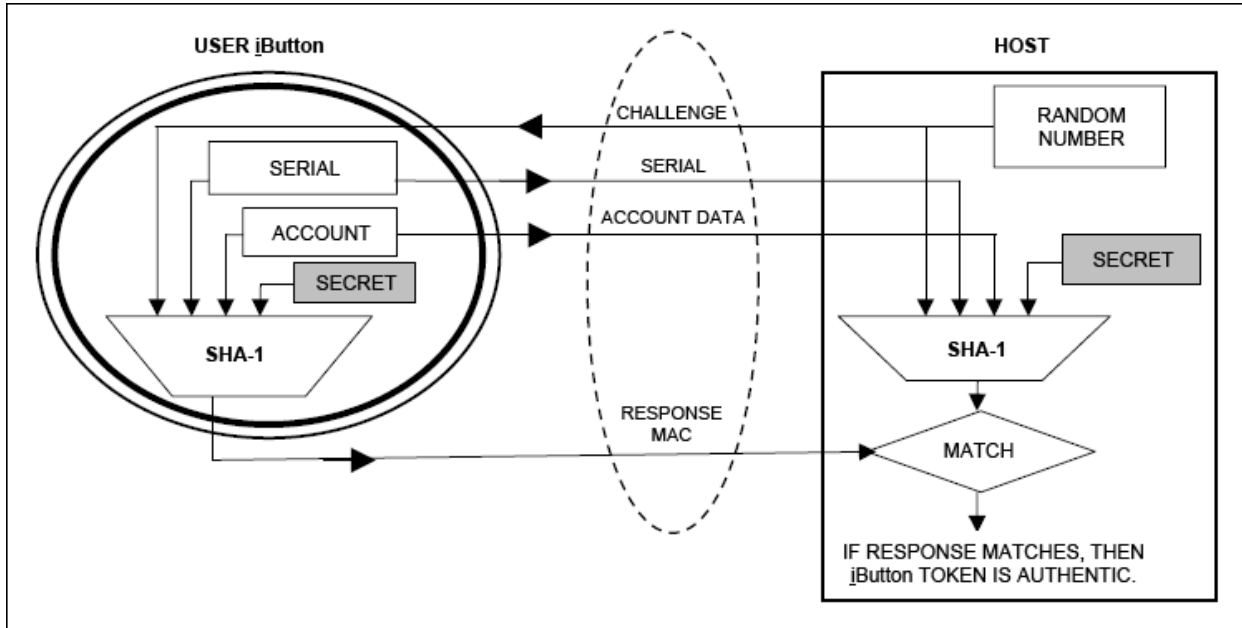

Figure 2. User Authentication

Example Demonstration

The 'RemoteAuth' example provided in the 1-Wire API for Java™ kit demonstrates challenge-and-response across a network. The kit can be found on Maxim's iButton page.

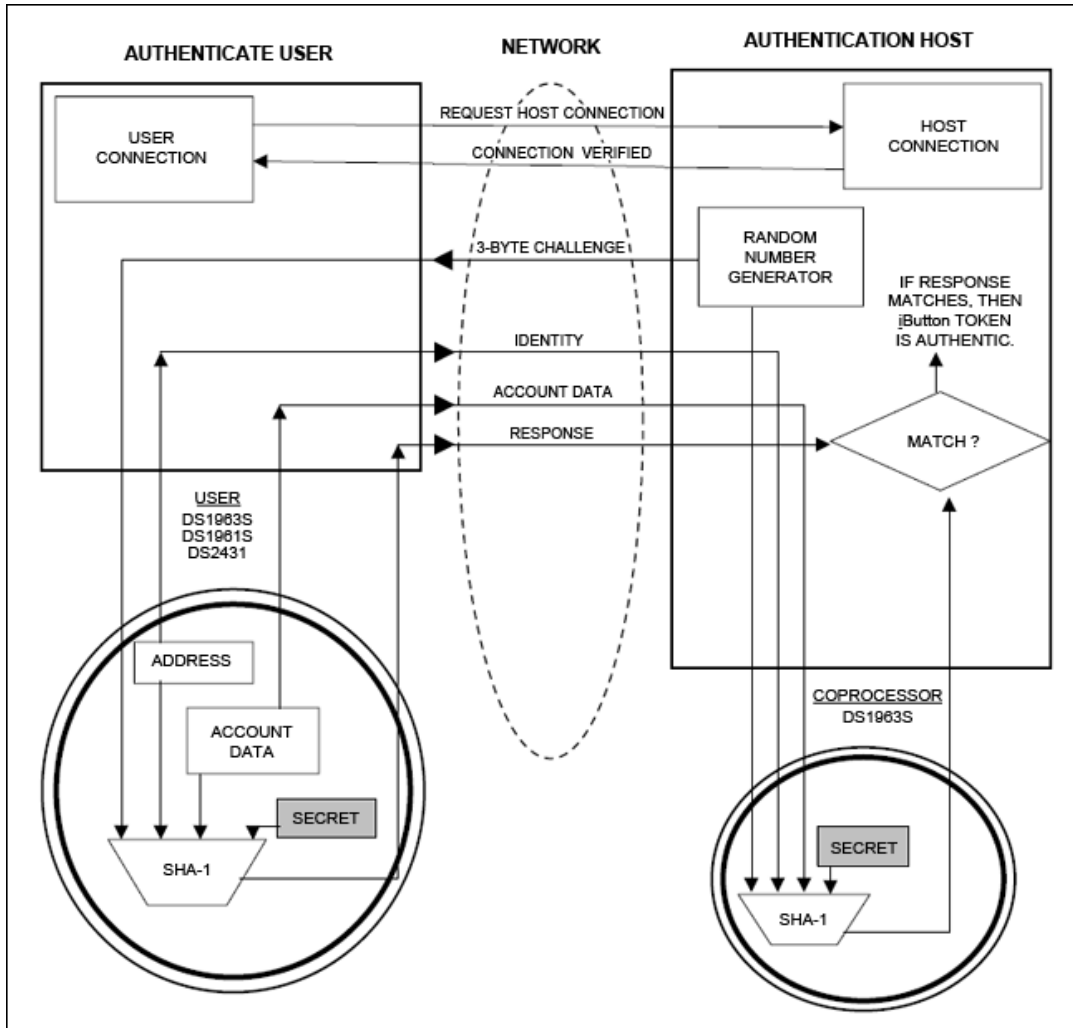See **Figure 3** for the layout and example data flow for the 'RemoteAuth' example.

Figure 3. RemoteAuth.

Conclusion

Challenge‑response is a secure way of protecting access to any privileged material. Many options for challenge‑response access were given. The most secure method is having a response that only the host can interpret and is different each time with a new random challenge. The Maxim 1‑Wire SHA devices prove to be excellent examples of devices to use for implementing challenge‑response.

Links to Useful Datasheets and Application Notes
1. DS1963S Datasheet
2. DS2432 Datasheet
3. DS1961S Datasheet
4. Application note 114, "1‑Wire File Structure"
5. Application note 151, "Maxim Digital Monetary Certificates"
6. Application note 156, "DS1963S SHA 1‑Wire API Users Guide"

1‑Wire is a registered trademark of Maxim Integrated Products, Inc.
iButton is a registered trademark of Maxim Integrated Products, Inc.

Java is a registered trademark and registered service mark of Oracle and/or its affiliates.

Related Parts		
DS1961S	1Kb Protected EEPROM ¡Button with SHA-1 Engine	
DS1963S	SHA ¡Button	
DS2432	1Kb Protected 1-Wire EEPROM with SHA-1 Engine	Free Samples

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

Application Note 190: http://www.maximintegrated.com/an190
APPLICATION NOTE 190, AN190, AN 190, APP190, Appnote190, Appnote 190
Copyright © by Maxim Integrated Products
Additional Legal Notices: http://www.maximintegrated.com/legal