A Virtual Storage System for Smart Home/Office Embedded Devices

데이 위슈*, 김영진**, 임효택*
*동서대학교 디자인 & IT 전문대학원
**동서대학교 컴퓨터공학과
e-mail : (ysthey,jkim)@dit.dongseo.ac.kr
htlim@dongseo.ac.kr

스마트 홈/오피스 임베디드 장치를 위한 가상 저장 시스템

Yu-Shu They*, Youngjin Kim**, Hyotaek Lim*
*Graduate School of Design and IT, Dongseo University
**Division of Computer Engineering , Dongseo University

Abstract

This paper proposes a virtual storage system for smart home/office embedded devices by utilizing AoE and HyperSCSI protocols. It relies on current Ethernet infrastructures and aims to provide a low-cost solution for the storage limitation problem on embedded devices. We choose AoE (ATA over Ethernet) and HyperSCSI in our implementation because they are relatively lightweight compared to other network storage protocols such as NFS(Network File System), CIFS(Common Internet File System) and iSCSI(internet Small Computer System Interface). The design and architecture of our proposed virtual system as well as the prototype demonstration are presented in this paper.

1. Introduction

With the advancement of embedded system and data communication technologies, several intelligent home/office equipments or smart devices have been invented and commercialized. The integration of these devices can be used to form an intelligent, automated home/office environment, referred as smart home/office.

According to [1], a smart home/office environment is built from the integration of the following systems:

1. Home/Office Automation System
2. Security System
3. Entertainment System
4. Communication System
5. Data System

People always intend to make these systems more powerful in order to create a “smarter” environment. This phenomenon is especially apparent in recent years, in which the demand for storage spaces has increases dramatically. People are getting interested in video, audio types of information rather then pure text data.

However, most of the embedded devices are still suffering from the insufficiency of storage spaces. Although the development of tiny storage technologies such as SD (Secure digital) memory card and CF (Compact Flash) memory card are growing rapidly, giving a larger storage spaces to an embedded device will definitely implies a higher cost to a smart home/office product. Besides, storing data in a distributed environment will also increase the difficulties for data management.

This paper presents an alternative solution for the storage limitation problem on embedded device using network storage approach. In contrast to the tiny storage approach, the data from the embedded devices are stored in network storage system. Thus, this approach requires all the related devices to be connected together under a LAN (Local Area Network) environment, so that the network storage system is able to be shared among these devices.

This approach consists of 3 major advantages.

1. Scalability – Storage devices are easier to be added in.
2. Manageability – Centralized data storage are easier to be managed.
3. Lower cost – Manufacturer cost for embedded devices are minimized.

The rest of this paper is organized as follows. In section 2, the system design, architecture, standards and technologies are discussed. A prototype demonstration based on the proposed system is presented in section 3 and finally we conclude the paper in Section 4.

2. System Design and Architecture

As stated in previous section, we assume that all embedded devices are connected together using Ethernet infrastructures to form a single LAN environment. Here, we briefly describe the standards and protocols that are use in this system.

From 1990s until now, Ethernet remains the most popular LAN technology. It has been standardized as IEEE 802.3 and widely used in everywhere. Due to its popularity, it is suitable to be used as a transport medium for storage data traffic.

In our proposed storage system, two block I/O network

storage protocols that rely on Ethernet layer are considered: AoE (ATA over Ethernet)[2] and HyperSCSI[3]. The AoE protocol is developed by Brantley Coile Company, designed for accessing ATA storage devices over Ethernet networks whereas the HyperSCSI protocol is developed by Data Storage Institute of Singapore to allow SCSI devices to be used over Ethernet connection.

Destination Address	Source Address	EtherType (0x88a2)	AoE Packet	CRC

Fig.1. Ethernet frame containing AoE packet.

Destination Address	Source Address	EtherType (0x889a)	HyperSCSI Packet	CRC

Fig.2. Ethernet frame containing HyperSCSI packet.

The ideas behind these two protocols are almost similar. Fig.1. and Fig.2. show the Ethernet frame format of AoE and HyperSCSI protocols. Disk I/O commands (ATA or SCSI) and the data blocks are encapsulate in the Ethernet frames and transmit over Ethernet network. As they are block level I/O protocols and does not require the present of file system layer at the target side, the file system layer overhead that introduced by the file I/O network storage protocols such as NFS (Network File System), and CIFS (Common Internet File System) has been avoided (as shown in Fig.3.)[4]. Unlike NFS, CIFS and iSCSI, AoE and HyperSCSI do not rely on TCP/IP layer and thus, there are no occurrence of network layer and transport layer overhead (as shown in Fig.4.).

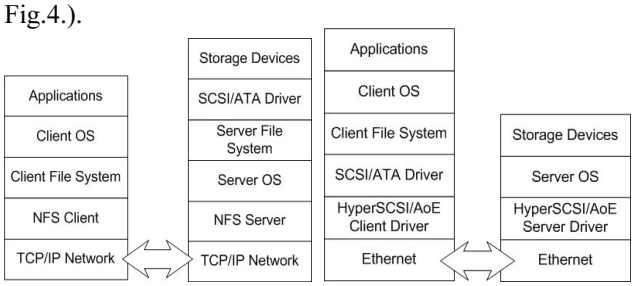

Fig.3. The “thin client” architecture of NFS system and the “fat client” architecture of AoE/HyperSCSI system

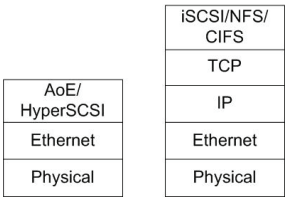

Fig.4. Less protocol overhead for AoE/HyperSCSI protocols

Fig.5. has shown an example of the smart home/office network configuration by using Ethernet infrastructure. AoE and HyperSCSI are co-exist protocols in this network storage environment. The virtual storage servers host the AoE and HyperSCSI services and export the appropriate ATA or SCSI devices over the Ethernet.

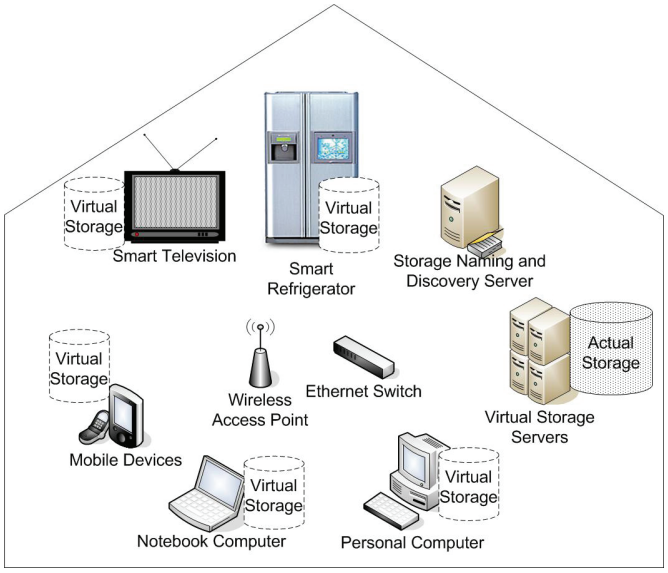

Fig.5. An example configuration for a smart home/office environment by using Ethernet.

As both AoE and HyperSCSI protocols are having their own naming scheme and storage discovery method, a unified external naming and discovery services are essential in order to improve the manageability and compatibility of the system. Furthermore, in the design of both AoE and HyperSCSI protocols, the states of the mounted device are managed by its mounting client; other client devices are unable to know the status of an exported device. Mounting a malfunctioned storage might cause harmful errors on the smart devices. Thus, a storage naming and discovery server is needed to keep track of the alias names and status of the exported devices.

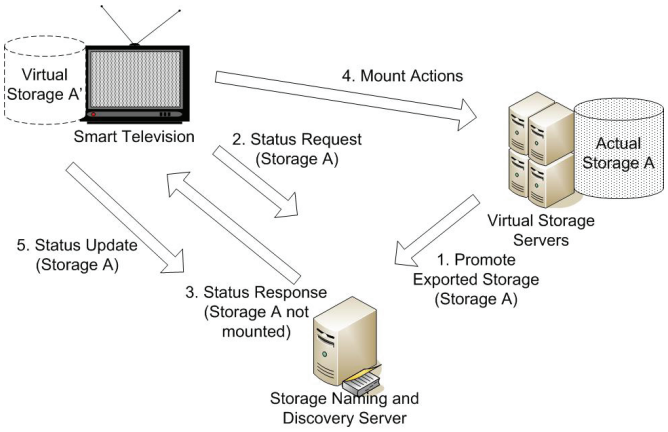

Fig.6. Storage discovery procedures.

Fig.6. illustrates the procedures of disk mounting by using the storage naming and discovery server. First, the information of the exported storage devices has to be published on the naming and discovery server. Before a client device is able to mount and use the exported storage device as its local storage, the status information of the particular storage has to be gathered from the naming and discovery server. If the exported storage is not in available state, the client device is not allowed to use it. This mechanism is uses to protect the storage against concurrency access and prevents the client system from mounting malfunction storage.

3. Prototype Implementation and Demonstration

Recently, various kinds of home/office products have burgeoned in market. Most of these devices come along with their own standards and platform. It is impossible for us to find a universal platform for our prototype implementation. In view of the rapid growth of embedded Linux technology, we decided to choose a PDA running embedded Linux operating system as our prototyping platform. The specifications of the above device are shown as below.

PDA	PDA Model	iPAQ 3800 Series
OS	Distribution	Familiar Linux 0.84
	Kernel	2.4.19
CPU	CPU Type	ARM v4l
	CPU Speed	200 MHz
Memory	ROM Size	64 Mb
	ROM Type	Flash
	RAM Size	64 Mb
	RAM Type	SD RAM

Fig.7. Embedded device specification

.

The open source implementation of AoE client module (version 1.7.0)[5] and HyperSCSI(version 20030506)[6] client module are cross-compiled and installed at the client side. On another side, AoE server module (vblade-14)[7] and HyperSCSI server module (version 20030506) are installed on an Intel Xeon Dual 800 MHz CPU machine with 512 Megabytes of RAM. The client embedded devices are attached to the Ethernet infrastructure via 802.11 Wireless LAN as shown in Fig.8.

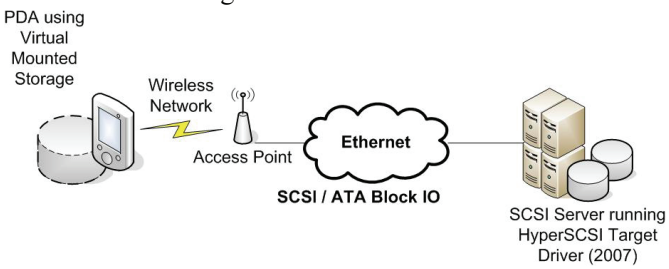

Fig.8. Testbed configuration.

As a result, PDA device is able to use the virtual storages as its local storage even though it is not directly attaches to the Ethernet. This is due to the fact that the WiFi (802.11) standard is fully compatible with the Ethernet. The following figures have shown that the exported ATA and SCSI storage devices are successfully mounted by the client embedded device.

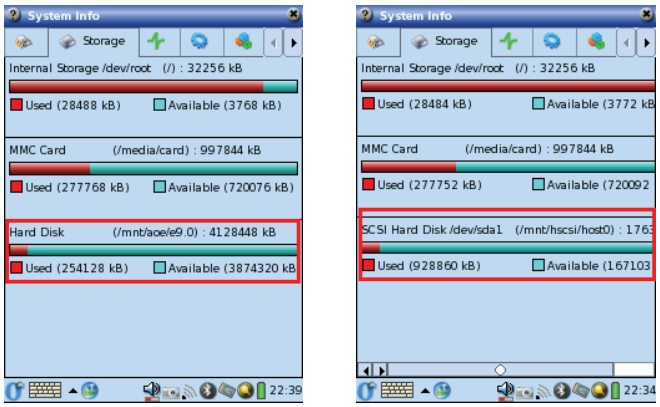

Fig.9. AoE device and HyperSCSI device are mounted as the local storage of the embedded device.

4. Conclusion

In this paper, a virtual storage system for smart home/office embedded devices by utilizing AoE and HyperSCSI protocols is proposed. Both or these protocols are relatively lightweight then other network storage protocols such as NFS, CIFS and iSCSI. By introducing an external naming and discovery server in the above network environment, we are able to form an AoE, HyperSCSI co-exist with the storage network that is suitable for embedded devices. The architecture of the proposed virtual storage system is presented and the usage of the proposed system is demonstrated on a prototyping device. In the near future, we will improve the manageability and reliability of this system by enhancing the functions of naming and discovery server.

Reference

[1] Dimitar Valtchev and Ivailo Frankov.: Service Gateway Architecture for a smart home. IEEE Communication Magazine (2002) 126-127

[2] Brantley Coile, Sam Hopkins.: The ATA over Ethernet Protocol (2004)

[3] Wilson Yong Hong Wang, Heng Ngi Yeo, Yao Long Zhu, Tow Chong.: Design and Development of Ethernet-Based Storage Area Network: IEEE (2004)

[4] Hyotaek Lim, Saebom Choi.: Design and Implementation of iSCSI-based Virtual Storage System for Mobile Health Care, HEALTHCOM 2005, Jun. 2005, PP 37-42

[5]AoE Initiator Project Page from Coraid, http://www.coraid.com/support/linux/

[6]HyperSCSI Project Page on Sourceforge.net, http://hyperscsi.sourceforge.net/

[7]AoE Tools Project from Sourceforge.net, http://sourceforge.net/projects/aoetools/