World Academy of Science, Engineering and Technology
International Journal of Electronics and Communication Engineering
Vol:1, No:9, 2007

Implementing High Performance VPN Router using Cavium's CN2560 Security Processor

Sang Su Lee, Sang Woo Lee, Yong Sung Jeon, and Ki Young Kim

***Abstract*—**IPsec protocol[1] is a set of security extensions developed by the IETF and it provides privacy and authentication services at the IP layer by using modern cryptography. In this paper, we describe both of H/W and S/W architectures of our router system, SRS-10. The system is designed to support high performance routing and IPsec VPN. Especially, we used Cavium's CN2560 processor to implement IPsec processing in inline-mode.

***Keywords*—**IP, router, VPN, IPsec.

I. Introduction

RECENT networks environment needs high performance router system which can handle network traffics in several giga bps rate. However, the traditional router systems based on S/W routing algorithm has obvious limitation in performance. So, network vendors favors using ASIC processor like network processors that is designed to lessen the burden of the system's host processor. However, for VPN, one of essential security requirements in recent traffic machines, most systems use S/W like FreeS/WAN or a kind of security processor which can handle IPsec protocol. Thus, they still has performance limitation due to the host processor's computing burden and frequent data copies between a physical network interface and kernel stack through PCI bus. Thus, inline processing architecture for IPsec is needed in favor of high speed VPN system.

In this paper, we describe our security router system, SRS-10, using Cavium's CN2560 processor which can processing packets with IPsec protocol in 10 giga bps rate, theoretically. S/W and H/W architecture of SRS-10 will be given to help reader's understanding in following sections.

II. The Architectures of SRS-10

A. H/W Architecture

The SRS-10 consists of following functional boards :

- ✓ SRMB (Secure Router Main Board) is the main board with Intel IXP 2800 network processor,
- ✓ SRNB (Secure Router Network Board) supports network interface such as Ample's Harrier A2510 for MAC and Marvell's Alaska 88E1145 for PHY,
- ✓ and SRCB (Secure Router Crypto Board) has exactly same as SRNB except that it has Cavium's CN2560 processor.

SRMB is connected to one of SRNB and SRCB. In detail, SRMB and SRNB can be combined for pure router system without IPsec functionality, while SRMB and SRCB for IPsec enabled router system. For convenience, we focus on SRMB and SRCB combination. Fig. 1 shows the H/W architecture of SRMB and SRCB.

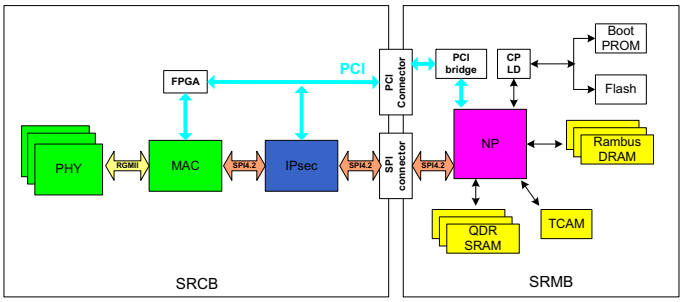

Fig. 1 The architectures of SRMB and SRCB

IXP 2800 has 16 of internal Microengines and an Xscale as host processor. Thus, drivers which give a data transfer mechanism for Xscale and Microengines are imported in Xscale.

CN2560 has 22 of internal Cipher Cores and 2 interfaces supporting SPI-4.2. Thus one of the interfaces is connected to MAC processor, while the other is connected to IXP 2800. Thus, packets received from PHY interfaces are multiplexed by MAC processor and delivered to CN25600 processor. Packets IPsec-processed by CN2560 are transferred to Microengines of NP through SPI-4.2 bus for router-related works such as lookup of routing table. CN2560 has also PCI/PCI-X interface, and it is connected to Xscale. Exception handling and configuration for CN2560 use the interface.

Readers can refer to [2] and [3] for detailed information of IXP 2800 and CN2560 processors, respectively.

B. S/W Architecture

S/W architectures for IXP 2800 and CN2560 processors should be discussed, but we focus on the latter because our interest is in IPsec VPN. However, [4] is a good reference for the S/W framework of IXP 2800 and readers can refer to it. Fig. 2 shows the H/W architecture of SRMB and SRCB.

Sang Su Lee is with with Electronics and Telecommunications Research Institute, Daejoen, Korea (phone: +82-42-860-1613; fax: +82-42-860-5611; e-mail: sangsu@etri.re.kr).

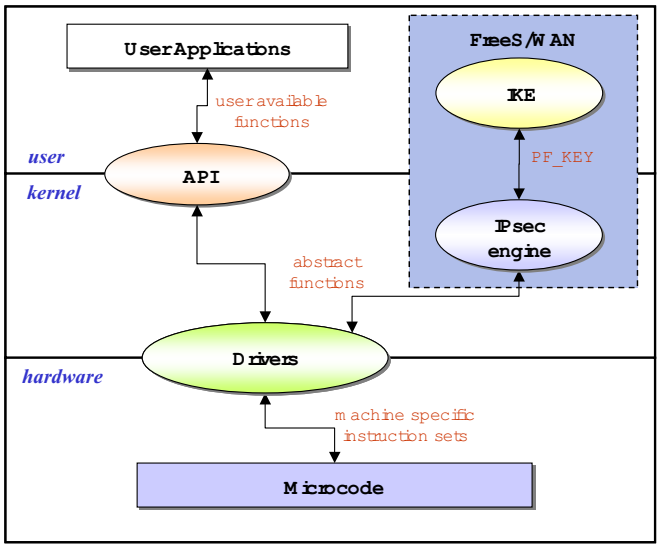

Fig. 2 S/W architecture for IPsec VPN

The Microcode is downloaded by one of user applications to CN2560 inside. FreeS/WAN is open source on internet, but we modified it to use hardware abstracted functions exported by drivers. One can ask that, if CN2560 itself can handle IPsec protocol, then what the reasons are that FreeS/WAN module exists. One of the answers is already given in previous section: for exception handling. Remember that our system has goal of processing IPsec protocol in almost 10 giga bps rate. For this, CN2560 must handle packets in flow-through or inline mode. Thus, many exceptions such as flagmented IP packets or reassembly of them and ICMP packet handling, can not be handled by the processor. These exceptions may not be frequent, so the CN2560 requests the host processor to handle them, and FreeS/WAN module installed in Xscale takes the requests. Another reason is for SA. IKE which negotiates SA with the other peers is very huge framework. Thus, IKE runs in user space generally, not implemented in H/W level. In our system, the release version of FreeS/WAN we use is 2.02. IPsec engine part of FreeS/WAN, KLIPS, is installed in kernel of Xscale as loadable module named "ipsec.o".

Drivers in Fig. 2 actually consists of two module: "n2_drv.o" and "eb2200.o".

"n2_drv.o" is the main driver module and raps CN2560 processor. It exports many functions which invoke the CN2560 specific instructions.

"eb2200.o" is the master of "n2_drv.o" in inter module communication between both of them. In detail, it imports many functions exported by "n2_drv.o" through inter module communication. It also exports two functions for "ipsec.o".

Fig. 3 shows the relationship of two drivers and ipsec module in terms of inter module communication.

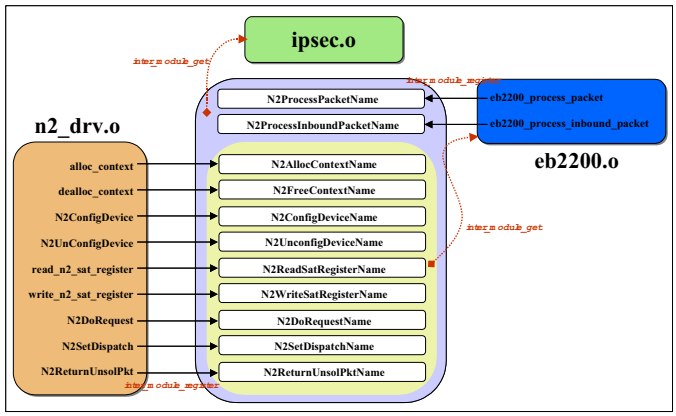

Fig. 3 S/W architecture for IPsec VPN

Note that integration of drivers for VPN and IXP 2800 is needed. Both of CN2560 and IXP 2800 are transparent to each other at the point of network device. Both processors view the same physical network interfaces, while the drivers treat the interfaces as different logical devices. That means, they handle the same packets from/to the same interfaces in different ways. For example, "eb2200.o" permits manual setting of MAC address of the peer, but the drivers of IXP 2800 doesn't. (In Linux OS, "arp" command with "-s" option sets MAC address of any node).

As of now, we have not done the integration, yet. Thus, optimization in S/W level is still in progress. However, simple performance test for IPsec VPN can be possible.

C. How to Process Packets in In-Line Mode

In this section, we give a detailed explanation of the way in which CN2560 processes the packets. Because "in-line mode processing " is our ultimate goal, we omits the description about "look-aside mode" processing of CN2560 processor.

Basically, CN2560 expects that a kind of CN2560 specific header information, named IRH (Input Request Header), be attached to an IP/IPsec packet. This header contains the opcode and other control information that describes how to process the packet. And, it also includes the complete context index which is used when the processor fetches appropriate SA information from external DRAM. Note that the uee of IRH for conveing data to/from the CN2560 is conditional, the device appends one of possible four values set in four internal registers, L3I_REQ_HDR0, L3I_REQ_HDR10, L3I_REQ_HDR20, and L3I_REQ_HDR30, shown in Fig. 4.

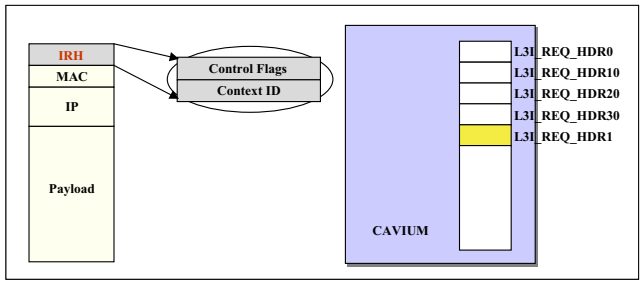

Fig. 4 IRH and related internal registers of CN2560

If an IPsec packet including AH or ESP header is delivered to the device, then SPI value included in the header is good enough to look up an SA from DRAM. The device refers to one of the registers (in this case L3I_REQ_HDR10) only for control flags as shown in Fig. 5.

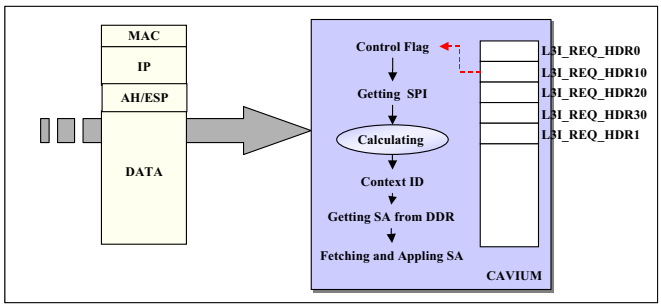

Fig. 5 IRH lookup for an IPsec packet

When the packet is non-IPsec packet, then the device has no way to lookup SA. "L3_REQ_HDR1" shown in figure 4 has the context value of an SA, Thus, the device depends on the registers for whole IRH information including control flags. However, it can't be applied in general applications because only one SA can be pointed by the register.

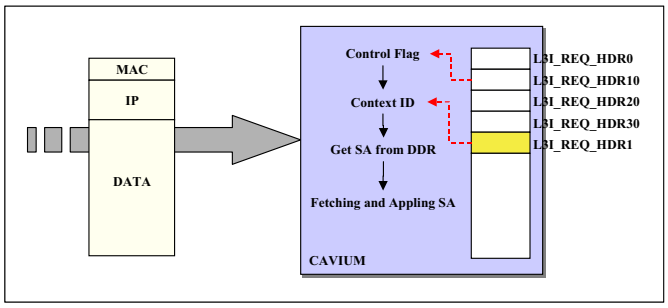

Fig. 6 IRH lookup for non IPsec packet

For practical in-line mode applications, Xscale or NP should attach IRH directly to the packet. As mentioned before, FreeS/WAN module in Xscale can do this. But, for 10 Gbps performance, Microengines connected through SPI-4.2 bus with CN2560 is more preferable. In this case, CN2560 must not refer to the registers discussed above, but parse IRH included in the packet.

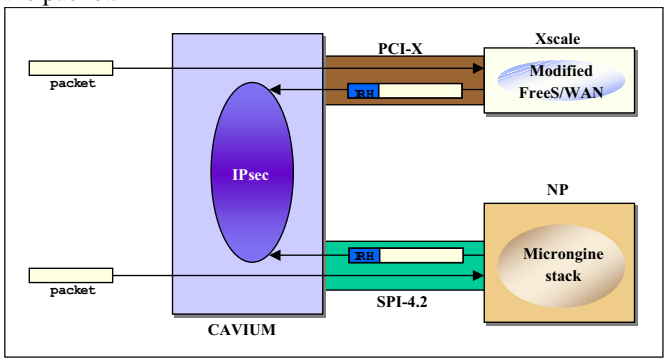

Fig. 7 IRH insertion to a packet by other processors

III. Test Results

For the performance test, we set up the test bed as shown in Fig. 8. For this test, we store one SA manually in DDR : 3DES-MD5/ESP/Tunnel mode. And, we set the context number of the SA in L3I_REQ_HDR1 register. That means the test was done in in-line mode. In the test, for packets with length of more than 64 bytes, almost 6 Gbps performance was obtained.

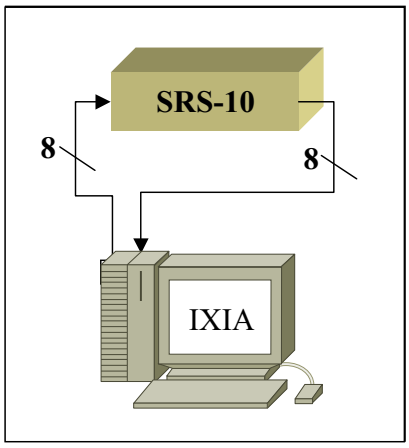

Fig. 8 Test Setup

We also did the test in look-aside mode through PCI-X bus. In our system, PCI-X bus runs in 66MHz with 64 bit length. Thus, one can expect almost 4 Gbps performance in theoretical. But, the test using PCI-X bus showed under 100 Mbps performance. We guess this low value is due to the computation burden of Xscale. And also, whole bandwidth of the bus was not occupied by the data transferring between CN2560 and Xscale.

IV. Conclusion

In this paper, we described both of H/W and S/W architectures of our router system, SRS-10. The system used both NP, Intel's IXP 2800 for general routing functions and a security processor, Cavium's CN2560 for high performance IPsec VPN. Our system is still in development in S/W aspects. However, simple performance test result was given.

References

[1] "Security Architecture for the Internet Protocol," RFC 2401, 1998
[2] Eric J. Johnson and Aaron R. Kunze, "IXP2400/2800 Programming," Intel Press, ch 2
[3] "Nitrox-II Security Processor Hardware Manual," Cavium Networks, Doc:CN2xxx-HM
[4] Bill Carlson, "Intel Internet Exchange Architecture and Applications," Intel Press, ch 1

Sang-Su Lee (M'01) became a Member(M) of **ETRI** in 2001. He received his BS and MS degrees in electronic engineering in 1999 and 2001, respectively, from Kyungpook National University, Korea. His research interests include optical information processing, optical security, and active network.