A virtualization-based method for analysis of Cisco IOS

Cheng ZENG, Shengli LIU, Ligen CHEN, Da XIAO
Zhengzhou Institute of Information Science and Technology
Zhengzhou, China
e-mail: chengceng777@126.com

***Abstract*—Traditional debugging and disassembly tools are much demanding to analyze as well as debug Cisco IOS. At the same time, analysis with these tools isn't performed in single-step. In addition, some other defects also contribute to its inaccuracy. This paper presents a dynamic debugging platform based on a router emulator. Subsequently, a new virtualization-based method is proposed on the basis of this platform to analyze Cisco IOS. With the control on virtual hardwares' running status and the trace of virtual router's running data, it's feasible to analyze IOS processes in single-step and accurately.**

Keywords-IOS process; virtualization; dynamic debugging platform; emulator

I. INTRODUCTION

Cisco Systems is the primary provider of routing and switching equipment for the Internet and most of corporate networks. On most of Cisco router & switch equipments, the Cisco IOS (originally called Internetwork Operating System) [3] is used, which is a non-open source operating system and standard ELF file. Cisco IOS image is loaded and decompressed to memory at ROMMON (ROM Monitor) mode. Cisco IOS does not use any internal protection mechanisms. Memory sections from one process are not shielded in any way from access by another process, and Cisco IOS makes heavy use of shared memory and global variables and flags accessible from any section of the code, there is no isolation among different processes. This leads to all kinds of attack, especially the buffer overflow exploit. In the past, successful exploitation of software vulnerabilities in Cisco IOS has been shown respectively by different researchers and groups, employing different techniques and basing of different assumptions. With that, attackers can reboot and control routers, executing arbitrary code in some versions of Cisco IOS. Router security is most essential to the whole network, but there is not any effective technique and tool for Cisco IOS security analysts. New tools and assumptions are requested to analyze security mechanisms, operating mechanisms and functionalities of Cisco IOS.

This paper will highlight a method for analysis of Cisco IOS based on virtualization. With heavy and appropriate use of breakpoint in the dynamic debugging and analysis platform which is based on virtualization technology, the specify IOS processes can be tracked step by step, and the address of the key memory data can be located. That work effectively when debugging IOS processes, and provide an effective method for the analysis of the security mechanisms, operating mechanisms and functionalities of Cisco IOS.

II. CISCO IOS ARCHITECTURE

Cisco IOS is a software used on most Cisco Systems routers and network switches which provides routing, switching, internetworking and telecommunication functions[3].A Cisco IOS image is a compressed file, and ELF file. Before a Cisco IOS runs on a router or switch, it was decompressed at boot-time as a single binary image. Because of the lack of process isolation, all the processes share the same memory space, and they do not have any protections, it means that if memory is corrupted in one of the running processes, any other Operating System component might be affected, so the tracking of the memory data should be at the time.

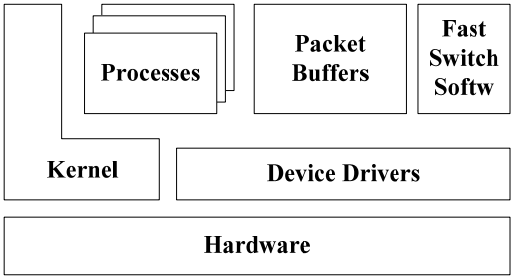

Figure 1. Cisco IOS process memory

A cooperative scheduler and a differentiation of process priority levels are implemented by the kernel. So that if a process runs for more than a predefined time it will be terminated by a watchdog, and have to relinquish their right to execute code and make a kernel call so that other processes can run, at the same time, higher priority processes can be re-arranged before lower priority ones. It is worth mentioning that the code stream get from the CPU In a period of time, could be composition of codes from a couple of processes. So tracking a specify IOS processes will be a difficult task. Though MMU (Memory Management Unit) have be used on their CPUs in recent versions of IOS, to set strict permissions on image sections, just like text section is marked with read-execute permissions.

These protection mechanisms [6]have been present on major Operating Systems for several years. Also recent versions of Cisco IOS perform a relocation of the main IOS image during boot process, which is considered as a rudimentary implementation of ASLR (Address Space Layout Randomization) [6].But the task of tracking a specify IOS processes is still a hard work.

Published by Atlantis Press, Paris, France.

III. DYNAMIC DEBUGGING PLATFORM

Cisco IOS runs on the PowerPC or MIPS platform, not the regular one but only the Cisco router or switch, by the analysis of the structure of IOS images. Cisco router models, used hardware emulation technology, must be so well for us comparing to the high cost of equipments.

A. Architecture of the debugging platform

Make sure each code of the IOS can be executed accurately, and just like be launched on a real one, is the core to build the virtual router. A sensible emulating of the underlying hardware of Cisco routers is requested, which contain all of the basic functions of them. Then, in virtual machine instance a remote debug module will be inserted, which is used to process all of the requests of user. At last, a remote debugger UI (User Interface) using debugging protocol should be done to communicate with the remote debug module.

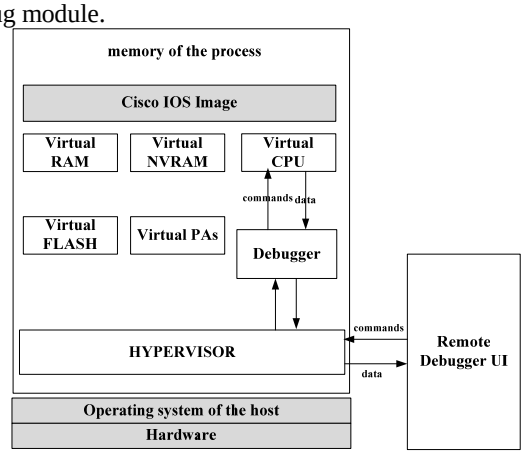

Figure 2. Architecture of the debugging syste

The common Intel platform is the selection of host operating platform, uses Linux operating systems, virtual hardwares of the virtual machine are a piece of data in the process memory space of the host or instructions of the execution process. First, the hardwares abstraction, virtual device management and user management interface can be achieved as the processes were created in the host operating system. Sufficient memory is applied in the process space of the virtual machine and emulated as the components in accordance with the different functions, as the virtual hardware be initialized when the virtual machine processes are up and running in the host operating system. Hypervisor runs parallel execution with the virtual CPU, it monitors the states of the virtual machine when it runs, and communicates with the user for the configuration and management of the virtual machine at the time.

B. Emulator of hardware

Virtual router hardware contains the CPU, storage devices, and input/output devices, classified according to the functional structure. The virtual router emulate hardwares according to the functional and structural characteristics of the hardwares.

1) Cpu

CPU is the core of the control and the operation of the computer, includes a arithmetic unit, a register, a controller ,a bus and so on, and its main function is to fetch instructions from memory, then put them into the instruction register, decod and execute the instructions, output the execution result to the router device[2].

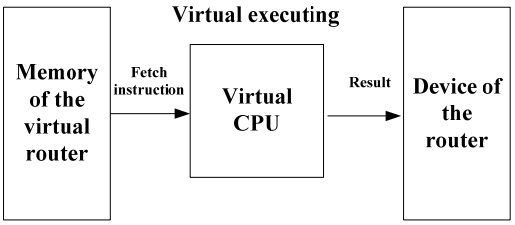

Figure 3. Running of the virtual CPU

2) Storage device

The router storage device includes RAM, Flash and NVRAM. The main function is used to store the IOS codes, routing tables, and system cache.In the design of Cisco router, all storage devices are mapped to a region of the virtual machine memory space. As usual,just a specified amount of memory in the process address space of the host is needed to be applied to create a virtual memory, it is mapped to the address space of the same size range in the memory space of the virtual machine,with the special equipment management structure to achieve the relationship management of the mapping.

3) Input/Output device

The input/output devices of the router provide receiving and forwarding functions, and provide user interaction management interface. NIC (Pas) is the key input/output device of a real router, the simulation of such component needs the support of the NIC of the host. Create a virtual NIC management structure in the virtual machine instance, set up a virtual NIC MAC address, bind its functions to the NIC of the host specify, with the related functions of the software pacp.

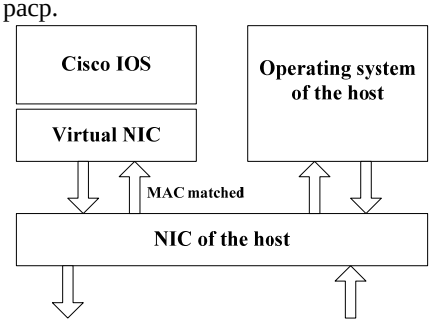

Figure 4. Virtual NIC

IV. METHOD OF DEBUGGING

The routing and switching gear of the router becomes increasingly complex, additional services besides simple packet forwarding it provides becomes more. Just as ftp, http and remote login. But most of them are not turned on when the router is launched, until users configure and turn them on.

Related information of each process will be added to the process table after been created, for its management.

```
Router#show process
CPU utilization for five seconds: 0%/0%; one minute: 2%; five minutes: 2%
 PID QTy       PC Runtime (ms)   Invoked   uSecs    Stacks TTY Process
   1 Csp 803E6FFC         312        45    6933 2608/3000   0 Load Meter
   2 M*          0         625       247  253010156/12000  0 Exec
   3 Lst 803CE93C         253        30    8433 5760/6000   0 Check heaps
   4 Cwe 803C3D8C           0         1       0 5604/6000   0 Chunk Manager
   5 Cwe 803D4218         132         2   66000 5584/6000   0 Pool Manager
   6 Mst 80324F24           0         2       0 5572/6000   0 Timers
   7 Mwe 8001BA10           0         2       0 5576/6000   0 Serial Backgroun
   8 Lsi 801A1D0C           0        22     362 5048/6000   0 ALARM_TRIGGER_SC
   9 Msi 80434104          12         9    1333 5508/6000   0 Environmental mo
  10 Lwe 8044ED90           0         6       0 5096/6000   0 ARP Input
  11 Mwe 80629E1C           0         2       0 5572/6000   0 DDR Timers
  12 Mwe 80647270           0         2       0 11580/12000 0 Dialer event
  13 Lwe 808179CC           4         2    2000 5600/6000   0 Entity MIB API
  14 Mwe 80020008           0         1       0 5612/6000   0 SERIAL A'detect
  15 Cwe 803D96A0           0         1       0 5604/6000   0 Critical Bkgnd
  16 Mwe 8039037C         220        55  400010432/12000  0 Net Background
  17 Lwe 8031A18C           4         5   80011436/12000  0 Logger
  18 Msp 8033BEB8         500       215    2325 5488/6000   0 TTY Background
  19 Msp 8038FA10        1114       221    5040 8684/9000   0 Per-Second Jobs
  20 Mwe 80098984           0         2       0 5528/6000   0 Hawkeye Backgrou
  21 Hwe 8038FBD0           0         1       0 5596/6000   0 Net Input
```

Figure 5. Process table

So much more information could be got from the list through the change, that help to locate the process specifically. From the parsing of instructions by the CPU, the code being executed in the router of the process can be extracted for debugging at the same time. And also, it cannot be determined whether the code been extracted currently is the code of the specify program's. So an instruction controller has been inserted into the debugging platform, tracking the running of CPU, through a group of standards, which are built through the information from the process table and parsing of the instruction in register.

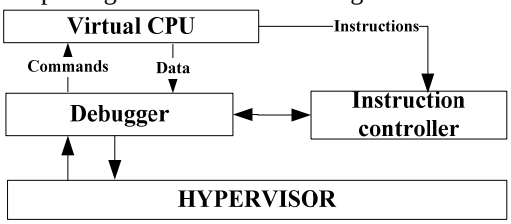

Figure 6. Modification of virtual CPU

With that the specified process code could be located and extracted accurately, and with the monitoring about the operation of the memory data in real-time. So the debugging of the specified process could be in single-step and accurately.

V. CONCLUSION

As Cisco router is an essential kind of equipment of network communication, it's security must be fully concentrated on. More and more researchers are focusing on Cisco routers. We present a dynamic debugging platform for Cisco IOS analysis, besides, a new method for the analysis of IOS process. This platform can track the IOS process more accurately. The method is not comprehensive. Further work may include research on embedded virtualization for system analysis.

REFERENCES

[1] Bollapragada V, Murphy C, White R. Inside Cisco IOS software architecture [M]. San Francisco:Cobden Press,2001.

[2] HAN Yu-xiang, LIU Sheng-li, LIU Long, SU Xiao-yan. An Analysis Method of Cisco IOS Communication Process [J]. Computer Engineering 2012, 38(18) 282-285

[3] Muñiz, Sebastian.Ortega, Alfredo. Fuzzing and Debugging Cisco IOS [C]. BlackHat EU 2011.

[4] Felix 'FX' Lindner. Cisco IOS Router Exploitation. Black Hat 2009.

[5] Gyan Chawdhary, Senior Consultant Varun Uppal, Senior Consultant. Cisco Shellcodes.Black Hat 2008.

[6] Anley, Heasman, Lindner, Richarte, The Shellcoder's Handbook: Discovering and Exploiting Security Holes, ISBN: 978-0470080238, Wiley, 2nd Edition, 2007.

[7] DU Hai,CHEN Rong. Full-virtualization-based Process Monitoring Method [J]. Computer Engineering 2009, 35(8) 88-90.

[8] Dynamips project, http://www.ipflow.utc.fr/index.php/Cisco7200 Simulator, Retrieved January 2011.

[9] VMWare virtualization software, http://www.vmware.com, Retrieved January 2011.

[10] Rootkits on Cisco IOS Devices, http://www.cisco.com/warp /public/707/cisco-sr-20080516-rootkits.shtml, Retrieved January 2011.