

## Extensible Network Security Device Built with the Amplify Solution

by John Lockwood, Washington University in Saint Louis

**This** year, the demand for Internet security has significantly increased. Internet connected hosts are now frequently attacked by malicious machines located throughout the world. Users are now bombarded daily with tens to hundreds of SPAM messages. Internet worms and viruses are now rapidly spreading from machine to machine. Networks can be made more secure by filtering traffic with content-aware, extensible network security devices. By actively dropping harmful packets and rate-limiting unwanted traffic flows, the damage caused by such attacks can be reduced.

While some types of attacks can be thwarted solely by examination of packet headers, other types of attacks – such as network intrusion, Internet worm propagation, and SPAM proliferation – require that network security devices process entire data content. Few existing devices have the capability to scan entire packet payloads. Of those that do, most are software-based and cannot process packets at the high-speed rates used by modern networks.

### System-on-Chip Firewall Technology

A single-chip, Internet security device has been developed that protects high-speed networks from present and future threats. The device was implemented on a single Xilinx Field Programmable Gate Array (FPGA) using the Synplicity<sup>®</sup> Amplify<sup>®</sup> Physical Optimizer™ solution. In order to protect networks against current threats, the baseline system parses Internet protocol headers, scans packet payloads, classifies traffic based on both the content and header files, and then performs per-flow queuing with selective transmission. In order to protect against future threats, the device allows new hardware components to be integrated as modules into FPGA hardware. The top-level architecture of the device is shown in Figure 1.

When a packet first enters the device, it is processed by a set of layered protocol wrappers. These wrappers segment and reassemble frames; verify and compute checksums; and read and write the headers of the Internet packet. Once the packet has been parsed by the wrappers, a payload scanner searches the entire content of the packet for keywords and regular expressions. A set of bits are set to identify which regular expressions were matched in the payload. Next, a Ternary Content Addressable Memory (TCAM) classifies the packet based on the value of the headers and the payload match bits. The result is a flow identifier (Flow ID) that is forwarded to an extensible module.

Extensible modules enable the network security device to perform additional, customized packet processing or functions. Extensible modules are implemented as a VHDL or Verilog component and are placed within a region of the FPGA device. Modules can examine, modify, add, queue, or drop packets as they pass through the system. A memory interface is provided to the module so that it can access off-chip Synchronous Dynamic Random Access Memory (SDRAM), if needed. The extensible modules can also modify the Flow ID, which is useful for implementing modules that aggregate flows together to protect a network from a Distributed Denial of Service (DDoS) attack.

The resulting flow identifier is output to a queue manager, which either drops the packet or schedules it for transmission on the outgoing network link. The packet itself is stored in a flow buffer. In order to buffer large amounts of traffic, the network security device stores packets in off-chip SDRAM. Once the packet scheduler determines it is time for the packet to be sent, data is read from SDRAM, processed by the layered protocol wrappers, and then transmitted on either a Gigabit Ethernet or SONET line card.

### Synthesis Flow

An automated design flow enables the network security device to easily be reprogrammed to implement additional functionality. As shown in Figure 2, the process begins when a new feature is uploaded to an on-line database.

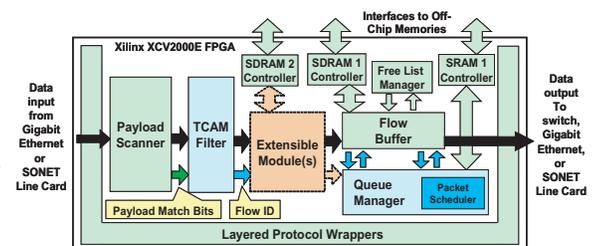


Figure 1: Architecture of the Reconfigurable Network Security Device

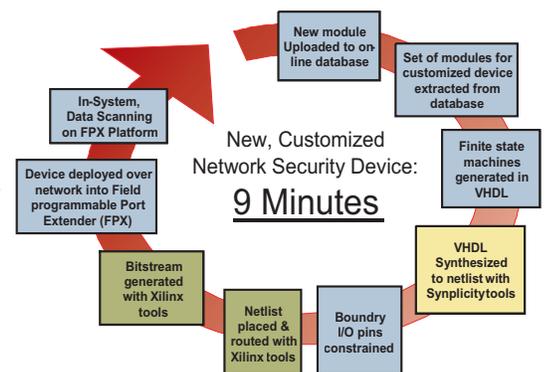


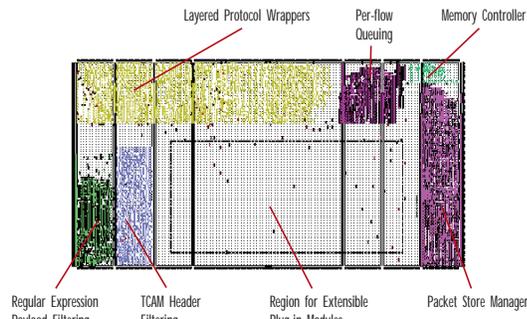
Figure 2: Automated design flow creates customized network security device

**“Extensible Network” continued from page 1**

that implement the desired feature set is extracted from the database. For the payload scanner module, VHDL code is automatically generated to create finite state machines that search for the regular expressions specified by the administrator. The resulting VHDL is then synthesizing into logic using Synplicity tools. In order to constrain where modules were placed, the Amplify solution is used to constrain the placement of logic. Likewise, the I/O are constrained to match the pads used by the FPGA on the implementation platform. This resulting netlist is placed and routed using FPGA tools and a bitstream is generated. The bit-stream is then deployed over the network to the Field programmable Port Extender (FPX) platform.

**Results**

Using an AMD Athlon 2400MP to perform all of the steps of the design flow shown in Figure 2, a complete network security device with the protocol wrappers, TCAM, flow buffer, and queue manager was synthesized in 9 minutes. Most of this time required to build the new circuit was used by the Xilinx tools to place and route the circuit. The logic of the network security device occupies 43% of the logic and 39% of the block RAMs of the Xilinx Virtex XCV2000E part used on the FPX. This circuit synthesized to operate at 62.5 MHz. Since each of these components process 32 bits of data in every cycle, the SOPC firewall achieves a throughput of  $32 * 62.5 \text{MHz} = 2$  Gigabits/second. A view of the resulting placed and routed FPGA is shown in Figure 3. Note that the center region of the chip was left available for insertion of extensible modules. This area of the chip can be used to implement new features.



**Figure 3:** FPGA Layout of the SOC Firewall on a Xilinx Virtex 2000E device. Amplify software was used to constrain the areas of each of the sub-components.

**In-system Testing**

The Field Programmable Port Extender (FPX) platform was used to implement the network security device and process real Internet traffic passing between a host and the Internet. The payload scanner was programmed to find regular expressions that contained signatures of computer viruses. The payload scanner examined the full payloads of all packets passing through the device and the TCAM was programmed to drop traffic that contained the virus. As expected, the virus-infected traffic was dropped and other traffic flows received their fair share of network bandwidth.



Synplicity, Inc.  
600 W. California Avenue  
Sunnyvale, CA 94086 USA  
Phone: (US) +1 408 215-6000  
Fax: (US) +1 408 222-0264  
[www.synplicity.com](http://www.synplicity.com)