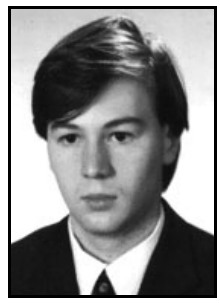
Grzegorz SUŁKOWSKI, Maciej TWARDY, Kazimierz WIATR
AKADEMICKIE CENTRUM KOMPUTEROWE CYFRONET AGH

Implementacja systemu bezpieczeństwa typu Firewall dla potrzeb sieci Ethernet w oparciu o układy reprogramowalne FPGA

Mgr inż. Grzegorz SUŁKOWSKI



Ukończył elektronikę na Wydziale EAIE Akademii Górniczo-Hutniczej w Krakowie. Od 2006 roku jest Konstruktorem Systemów Obliczeniowych w dziale Archiwizacji i Bezpieczeństwa Danych w ACK CYFRONET AGH. Jego zainteresowania skupiają się wokół algorytmów obliczeniowych oraz ich realizacji w logice reprogramowalnej.

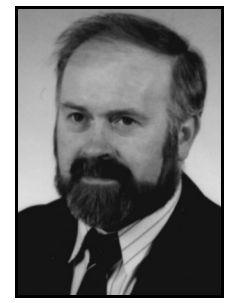
e-mail: Grzegorz.Sulkowski@cyfronet.pl

Mgr inż. Maciej TWARDY

Ukończył elektronikę na Wydziale Elektrotechniki, Automatyki, Informatyki i Elektroniki Akademii Górniczo-Hutniczej w Krakowie. Od 2005 roku kieruje Działem Archiwizacji i Bezpieczeństwa Danych w Akademickim Centrum Komputerowym CYFRONET AGH. Jego zainteresowania związane są z szeroko pojętą informatyką oraz projektowaniem układów cyfrowych w oparciu o logikę reprogramowalną.

e-mail: Maciej.Twardy@cyfronet.pl

Prof. dr hab. inż. Kazimierz WIATR



Studia AGH Kraków (1980), dr nauk technicznych (1987), dr habilitowany (1999) i profesor (2002). Profesor zwyczajny na Akademii Górniczo-Hutniczej oraz Dyrektor Akademickiego Centrum Komputerowego Cyfronet AGH. Prowadzone prace badawcze dotyczą komputerowego sterowania procesami, systemów wizyjnych, systemów wieloprocesorowych, układów programowalnych, rekonfigurowalnych systemów obliczeniowych i sprzętowych metod akceleracji obliczeń.

e-mail: wiatr@agh.edu.pl

Streszczenie

W artykule omówiono prace badawcze dotyczące budowy sprzętowego systemu bezpieczeństwa typu Firewall dla ochrony zasobów w sieci Ethernet. Implementacja takiego systemu w układach programowalnych FPGA z jednej strony uniemożliwi jakiekolwiek włamania do systemu bezpieczeństwa, z drugiej natomiast rekonfigurowalność układu FPGA pozwoli na łatwe modyfikacje tego systemu, w tym także modyfikacje zdalne. Opracowywany system bezpieczeństwa typu Firewall, implementowany w układzie programowalnym FPGA, wpisuje się w aktualny nurt badań światowych nad budową zasobów rozbudowanych elementów bibliotecznych typu IP Cores, przeznaczonych do projektowania rozbudowanych systemów obliczeniowych.

Słowa kluczowe: systemy bezpieczeństwa informatycznego, układy programowalne, języki opisu sprzętu, Ethernet, firewall.

Implementation of the Ethernet Firewall security system in FPGA programmable logic

Abstract

In this document authors discuss current stage of their work focused on firewall security system implemented in FPGA technology and dedicated for Ethernet LAN. The FPGA technology ensures high security level and can protect from hackers attack. On the other hand, the FPGA technology allow in simple way to change the firewall configuration and settings via the remote reconfiguration mechanisms. Authors hope that designed security system will be widely used as an IPCore library element in large computing systems.

Keywords: information security systems, programmable logic, hardware description language, Ethernet, firewall.

1. Wstęp

Dynamiczny rozwój systemów teleinformatycznych w ostatniej dekadzie doprowadził do sytuacji, w której trudno dziś wyobrazić sobie funkcjonowanie współczesnej organizacji lub firmy bez posiadania przez nią systemów komputerowych, korzystających z zasobów sieci lokalnych czy też publicznych. Systemy informatyczne stały się integralną, w niektórych sytuacjach nieodzowną, częścią infrastruktury organizacji, obejmując zasięgiem swego działania coraz to nowe obszary o znaczeniu strategicznym. Przetwarzane przez nie informacje mają w wielu przypadkach charakter niejawny, nie powinny więc być ogólnie dostępne. Utrzymanie poufności informacji staje się bardzo trudnym zadaniem, głównie ze względu na wzrastające wykorzystanie teletransmisji danych, łączenie rozrzuconych geograficznie jednostek organizacji za pomocą sieci rozległych WAN (ang. *Wide-Area Network*), przy jednoczesnym eksploatowaniu zasobów sieci publicznych, przede wszystkim Internetu. Również protokoły komunikacyjne np. TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*), nie posiadają standardowo wbudowanych mechanizmów bezpieczeństwa, takich jak zapewnienie autentykacji, integralności czy też poufności transportowanych informacji. Konieczne stało się w tej sytuacji opracowanie odpowiednich środków ochrony, pozwalających na ograniczenie dostępu do krytycznych zasobów (zgodnie z założeniami polityki bezpieczeństwa organizacji). Systemy bezpieczeństwa, realizujące powyższe zadania, od początku swego istnienia do dziś, opierają się w zdecydowanej większości na rozwiązaniach programowych. Realizowane przez nie funkcje ochrony danych wykonywane są przez specjalne oprogramowanie uruchamiane na platformach sprzętowych ogólnego przeznaczenia. Tylko w nielicznych wypadkach tworzone są rozwiązania dedykowane: specjalnie zaprojektowany sprzęt wraz z niezbędnym oprogramowaniem, co jednak znacznie podnosi koszty całego urządzenia. Standardowe podejście do budowy systemów bezpieczeństwa w oparciu o oprogramowanie posiada jednak wiele wad, do których należy przede wszystkim duża podatność na próby naruszenia bezpieczeństwa, związana m. in. z wykorzystywaniem błędów systemów operacyjnych, stanowiących platformę dla właściwego mechanizmu zabezpieczającego. Wzrastające zapotrzebowanie na wydajność, wynikające z przetwarzania coraz większej ilości informacji, jest w przypadku takich rozwiązań rekompensowane poprzez stosowanie coraz szybszych procesorów ogólnego przeznaczenia GPP (ang. *General Purpose Processors*).

Autorzy, w ramach prowadzonego projektu badawczego, chcą wykazać, iż istnieje odmienna droga służąca budowie wydajnego i stabilnego systemu bezpieczeństwa typu Firewall, polegająca na jego implementacji w układach logiki reprogramowalnej FPGA. Takie podejście pozwoli na wyeliminowanie składników systemu operacyjnego oraz oprogramowania. Reguły bezpieczeństwa (polityka bezpieczeństwa) pozostaną jedynym elementem systemu istniejącym poza strukturą programowalną. Brak komponentów software'owych wykluczy możliwość włamań poprzez uruchamianie obcego kodu, przejmowanie uprawnień, itp. Analiza ruchu sieciowego na poziomie sprzętowym pozwala oczekiwać znacznego zwiększenia poziomu wydajności w stosunku do rozwiązań konwencjonalnych.

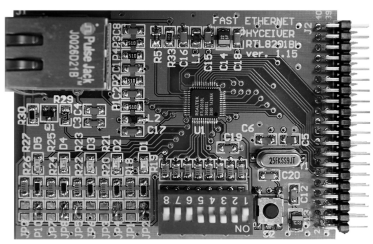
Sam proces realizacji tego zadania to przedsięwzięcie bardzo złożone, wymagające od projektantów dużej wiedzy z zakresu funkcjonowania sieci typu Ethernet oraz systemów bezpieczeń-

stwa informatycznego, a przede wszystkim bogatych doświadczeń w projektowaniu układów z wykorzystanie języków opisu sprzętu. Całość prac projektowych została podzielona na kilka etapów, m.in.:

a) przygotowanie odpowiedniego stanowiska badawczego, obejmującego platformę testową z układem FPGA oraz dedykowane do niej karty interfejsów sieciowych,
b) modularyzację Firewall'a pozwalającą na wyodrębnienie bloków funkcjonalnych, które będą kolejno implementowane w układzie FPGA,
c) opracowanie odpowiednich procedur testowych, weryfikujących poprawność funkcjonowania zaimplementowanych bloków.

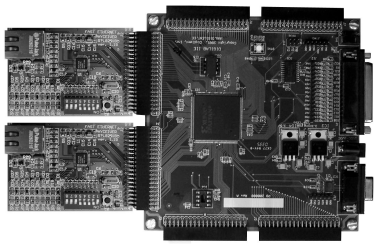
2. Platforma badawcza

Jako platformę badawczą, w pierwszym okresie prowadzenia prac projektowych, wykorzystano płytę z układem FPGA Xilinx Spartan2E produkcji firmy Digilent. Niezbędne było jej doposażenie o minimum dwie karty sieciowe w standardzie FastEthernet. Zostały one zaprojektowane i wykonane w ramach realizacji projektu w oparciu o układ PHY (ang. *Physical Control Layer)* RTL8201BL produkcji firmy Realtek. Odpowiada on za kodowanie, dekodowanie i synchronizację na poziomie nośnika danych, pozwalając na realizowanie transmisji z prędkościami 10 lub 100Mb/s przy zachowaniu pełnej zgodności ze standardami IEEE 802.3/802.u. Od strony funkcjonalnej istotna jest możliwość pełnej parametryzacji karty, zarówno w sposób manualny (za pomocą zestawu przełączników DIP switch), jak również sprzętowo poprzez szeregowy interfejs MDC/MDIO. Karta wyposażona została w zestaw diod LED sygnalizujących kluczowe parametry, m.in.: istnienie poprawnego stanu łącza (link), aktualną prędkość transmisji 10 lub 100Mb/s, wystąpienie kolizji na łączu. Interfejs wyjściowy karty dostosowano do specyfikacji portów płyty uruchomieniowej firmy Digilent.



Rys. 1. Karta interfejsu sieciowego Ethernet 10/100 Mb/s
Fig. 1. Ethernet 10/100 Mb/s Network Interface Card

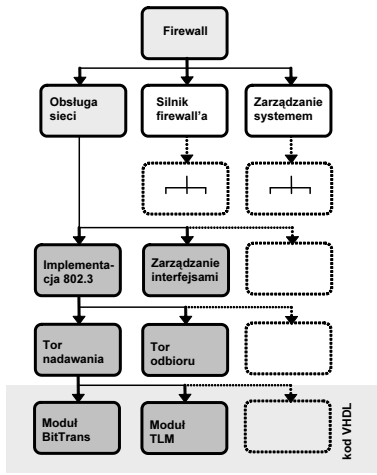
Pełny zestaw badawczy, obejmujący dwa interfejsy sieciowe oraz płytę Digilent, przedstawiono na rys. 2.



Rys. 2. Kompletna platforma badawcza
Fig. 2. Development platform

3. Modularyzacja Firewalla

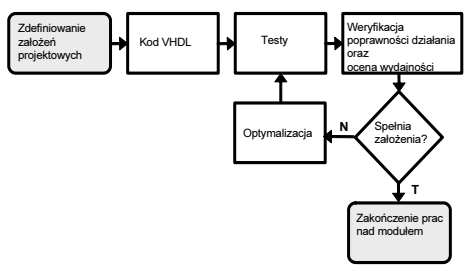
Ze względu na swą złożoność, Firewall został poddany dekompozycji na mniejsze bloki funkcjonalne. Proces ten odbywał się zstępująco w kilku turach: począwszy od ogólnego, na najwyższym poziomie hierarchii, do bardzo szczegółowego w obrębie konkretnych modułów. Takie podejście usprawnia realizację projektu, pozwalając na podział prac pomiędzy zespół projektowy, przy zachowaniu wzajemnej kompatybilności tworzonych modułów, umożliwiając równocześnie efektywną weryfikację osiągniętych rezultatów. Aktualnie realizowany fragment struktury blokowej projektu Firewall'a przedstawiono na rys. 3.



Rys. 3. Fragment schematu blokowego Firewall'a
Fig. 3. Fragment of the Firewall's block diagram

4. Zaawansowanie prac badawczych

Po zakończeniu pierwszego etapu, związanego kompletowaniem stanowiska badawczego oraz modularyzacją systemu bezpieczeństwa, autorzy przystąpili do realizacji zasadniczych elementów projektu. Przyjęte na wstępie założenie o spodziewanych korzyściach płynących ze sprzętowej implementacji systemu bezpieczeństwa, zmusza projektantów do położenia szczególnego nacisku nie tylko na kwestie zapewnienia żądanej funkcjonalności, ale również na optymalizację parametrów wydajnościowych poszczególnych komponentów. Sytuacja ta dotyczy modułów na każdym poziomie hierarchii projektu: począwszy od mechanizmów sterujących komunikacją sieciową a skończywszy na samym silniku Firewall'a przetwarzającym reguły bezpieczeństwa. Każdy z tych elementów wnosi pewne opóźnienia w ścieżce przepływu informacji, co sumarycznie może doprowadzić do sytuacji, w której finalne rozwiązanie będzie posiadać gorsze parametry, niż w przypadku tradycyjnego podejścia software'owego. W praktyce proces tworzenia modułu, uwarunkowany opisywanymi wcześniej czynnikami, zamyka się w cyklu projektowym, bazującym na przedstawionej na rys. 4 sekwencji działań.



Rys. 4. Cykl projektowy pojedynczego modułu
Fig. 4. Single module project cycle

Każdy z wyodrębnionych modułów musi wpierw zostać opisany z wykorzystaniem języka VHDL (ang. *Very High Speed Integrated*

*Circuits **H**ardware **D**escription **L**anguage*). Następnie powstały kod jest poddawany symulacji i testom z wykorzystaniem odpowiednich narzędzi projektowych, a otrzymane wyniki zostają skonfrontowane z wymaganiami projektowymi. Jeżeli funkcjonalność, bądź otrzymane parametry wydajnościowe odbiegają od przyjętych założeń konieczne jest przeprowadzenie niezbędnych korekt w opisie i ponowne przetestowanie całego modułu.

Pierwszą z realizowanych w ramach projektu badawczego grupą funkcjonalną jest część odpowiedzialna za obsługę komunikacji sieciowej (zaznaczona ciemniejszym kolorem na schemacie blokowym na rys. 3). Jej najistotniejszy element to implementacja standardu sieci Ethernet IEEE 802.3 [2], niewątpliwie najbardziej popularnego obecnie sposobu komunikacji w obrębie sieci lokalnych LAN (ang. *Local Area Network*). Autorzy postawili sobie za cel zachowanie pełnej zgodności z zaleceniami standardu, obejmującymi również poprawki dotyczące obsługi transmisji z prędkościami 1Gb/s. Ten etap prac jest niezwykle istotny ze względu na wpływ na końcową wydajność całego systemu. Wynika to ze sposobu funkcjonowania interfejsu sieciowego PHY: odpowiada on jedynie za dopasowanie do standardu warstwy fizycznej kanału transmisyjnego, począwszy od interfejsu MII lub GMII (ang. *Media Independent Interface, Gigabit Media Independent Interface*). W klasycznym podejściu przygotowania pakietów i ramek do transmisji oraz analizy ich zawartość po odebraniu danych przez interfejs sieciowy dokonuje procesor. Przekłada się to na znaczy wzrost obciążenia, a tym samym na spadek efektywności całego systemu. Zjawisko takie możemy zaobserwować nawet w przypadku standardowych komputerów osobistych. O ile nie jesteśmy praktycznie w stanie dostrzec wpływu obsługi komunikacji sieciowej na obciążenie przy komunikacji z prędkością 100Mb/s, co ma związek z bardzo dużą wydajnością dostępnych obecnie procesorów, to przy 1Gb/s jest on już wyraźnie dostrzegalny. Przeniesienie warstwy 2 i 3 modelu ISO/OSI do układu FPGA stwarza zatem realne szanse na przyspieszenie przetwarzania danych.

Proces implementacji standardu IEEE 802.3 został podzielony na trzy główne części:

- moduł TX – tor nadawczy,
- moduł RX – tor odbiorczy,
- moduł zarządzania (MII) – komunikacja pomiędzy warstwą 1 oraz 2 modelu ISO/OSI w standardzie 802.3.

Prace projektowe rozpoczęto od modułu MII, zapewniającego mechanizmy komunikacji pomiędzy warstwami 1 oraz 2 modelu ISO/OSI. Dostarcza on informacji o aktualnych parametrach warstwy fizycznej sieci pochodzących z układu PHY do modułów realizujących nadawanie oraz odbiór. Pozwala również zmieniać konfigurację wewnętrznych rejestrów PHY'a, a tym samym modyfikować w razie potrzeby właściwości kanału (medium) transmisyjnego. Ponieważ omawiany moduł nie posiada dużej złożoności funkcjonalnej, przyjęto by implementacja była możliwie zwarta i jednocześnie nie zajmowała dużej liczby zasobów układowych. Pozwoliło to na wykorzystanie pojedynczej maszyny stanów uzupełnionej o kilka układów pomocniczych, jak np. dzielniki częstotliwości. Już na tym etapie koniecznym stało się opracowanie wewnętrznych mechanizmów pozwalających na efektywną wymianę informacji pomiędzy poszczególnymi elementami całego systemu. Z tego względu zdecydowano się na wydzielenie w układach FPGA specjalnych obszarów wewnętrznej pamięci BRAM z przeznaczeniem ich na podręczne bufory transmisyjne oraz utworzenie wspólnej tablicy deskryptorów trybu pracy w obrębie jednego interfejsu sieciowego.

Moduł nadawczy TX, na podstawie informacji o aktualnych parametrach medium transmisyjnego (m.in. tryb i prędkość nadawania, zajętości medium) pochodzących z modułu MII oraz układu PHY, dokonuje wyboru właściwego sposobu realizacji transmisji. Najprostszym przypadkiem jest praca w pełnym duplex'ie, kiedy możliwe jest równoczesne nadawanie i odbieranie danych. Nie występują wówczas kolizje, moduł nadawczy nie monitoruje sygnału zajętości medium a realizuje jedynie funkcje niezbędne do właściwego przygotowania ramki Ethernetowej dla układu PHY (generuje wymagane pola ramki, m.in. preambułę, SFD, oraz wylicza sumę kontrolną CRC dla wysyłanych danych). Najbardziej złożony przypadek, to praca w trybie half-duplex. Moduł nadawczy funkcjonuje wówczas zgodnie z algorytmem CSMA/CD (ang. *Carrier Sense Multiple Access with Collision Detection*) nasłuchując nieustająco sygnał zajętości medium transmisyjnego (*carrier sense*). W przypadku wystąpienie kolizji podczas nadawania ramki, transmisja jest podtrzymywana jeszcze przez pewien czas – jest to tzw. wymuszanie kolizji, zwiększające prawdopodobieństwo wykrycia kolizji przez wszystkie stacje funkcjonujące w danej sieci. Transmisja może być ponowiona dopiero po pewnej zwłoce (tzw. backoff).

W przypadku modułu odbiorczego RX sposób funkcjonowania jest również ściśle zależny od aktualnego trybu pracy medium transmisyjnego. Analogicznie, jak to miało miejsce w przypadku modułu TX, monitorowane są informacje przekazywane przez MII oraz PHY i na tej podstawie wybierany jest bieżący algorytm pracy. Głównym zadaniem realizowanym przez moduł RX jest dekompozycja poszczególnych pól ramki Ethernetowej z odebranych od interfejsu PHY nibli (połówek bajtów) danych oraz wyliczanie sumy kontrolnej CRC celem weryfikacji poprawności transmisji. Wyselekcjonowane pola ramki są zapisywane w buforze odbiorczym z wyrównaniem 32 bitowym. Rozwiązanie takie ma na celu przygotowanie danych do wygodnego przetwarzania w silniku Firewall'a, w którym zdecydowana większość operacji będzie przeprowadzana na słowach o długości 32 bitów (adresy IP źródłowy i docelowy, maski podsieci źródłowej i docelowej).

Zajętość zasobów układu FPGA dla poszczególnych modułów na obecnym etapie realizacji projektu została przedstawiona w tabeli 1.

Tab. 1. Wykorzystanie zasobów układu FPGA Xilinx Spartan 2E
Tab. 1. Xilinx Spartan 2E device utilization summary

Zasoby układowe	Moduł MII	Moduł RX	Moduł TX
Slices	9%	8%	11%
GCLKs	25%	25%	50%

5. Podsumowanie

Moduły wykonane na obecnym etapie prac badawczych realizują operacje związane z transmisją i odbiorem danych na poziomie ramek Ethernetowych. W kolejnym etapie opracowane zostaną bloki odpowiedzialne za zarządzanie modułami RX i TX oraz za analizę pakietów IP, w tym pozyskiwanie informacji niezbędnych do weryfikowania reguł bezpieczeństwa.

Praca naukowa finansowana ze środków na naukę w latach 2006-2008 jako projekt badawczy.

6. Literatura

[1] K. Wiatr: Akceleracja obliczeń w systemach wizyjnych, Wydawnictwa Naukowo-Techniczne, Warszawa 2003.
[2] 802.3 IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks, 2002.
[3] K. Skahill: Język VHDL. Projektowanie programowalnych układów logicznych. Wydawnictwa Naukowo-Techniczne, Warszawa 2001.
[4] W. Stallings: Ochrona danych w sieci i intersieci. W teorii i praktyce. Wydawnictwa Komunikacji i Łączności, Wydawnictwa Naukowo-Techniczne, Warszawa 1997.
[5] XILINX: The Programmable Logic Data Book. Xilinx Inc. 1999.
[6] D.E Commer: tom 1: "Sieci komputerowe TCP/IP. Zasady, protokoły, architektura", WNT 1997.
[7] W. Richard Stevens: Biblia TCP/IP, tom 1. Protokoły. Wydawnictwo RM, Warszawa 1998.

Artykuł recenzowany