

Piotr GAJ

Politechnika Śląska, Instytut Informatyki

Józef OBER

Instytut Informatyki Teoretycznej i Stosowanej PAN

FIREWALL++ DO ZASTOSOWAŃ W SYSTEMACH PRZEMYSŁOWYCH

Streszczenie. Współczesne informatyczne systemy przemysłowe wymagają z jednej strony zdeterminowanej w czasie sieciowej obsługi informacji, z drugiej zaś niezbędna okazuje się konieczność współpracy takiego systemu z abonentami pracującymi w intersieci. Istotnym zagadnieniem staje się konstrukcja oprogramowania pośredniczącego w wymianie informacji pomiędzy intersiecią a siecią systemową. W artykule zaproponowano zastosowanie do tego celu specjalistycznej aplikacji ściany ogniowej *Firewall++*, pracującej na abonencie pośredniczącym.

Słowa kluczowe: TCP/IP, intersieci, sieci przemysłowe, determinizm, firewall.

FIREWALL FOR USING IN INDUSTRIAL SYSTEMS

Summary. In present-day industry computer systems require on one hand time deterministic data service and on the other hand there is a necessity of cooperation between the system and the internetwork subscribers. The key issue is construction of the software, which mediates in data exchange between the internetwork and the system network. In the article it is proposed to use special application of firewall called *Firewall++* working on transitional subscriber for this purpose.

Keywords: TCP/IP, intersieci, industrial networks, determinism, firewall.

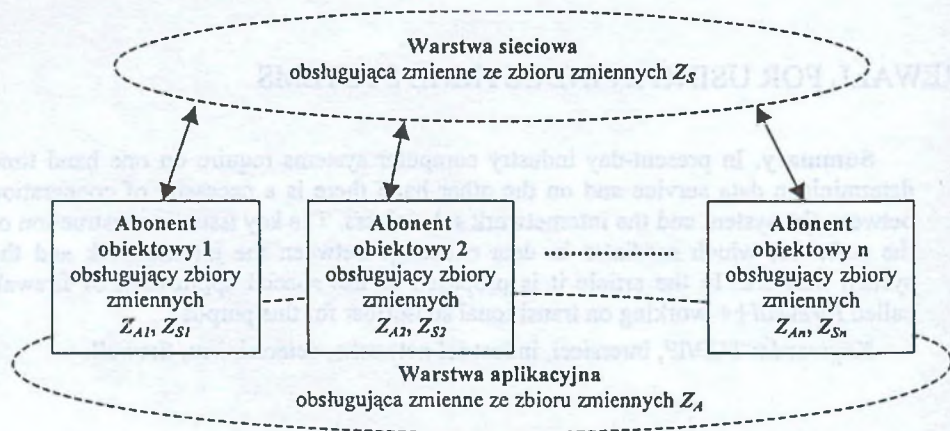
1. Wstęp

Współcześnie coraz częściej przy projektowaniu informatycznych systemów przemysłowych stawia się wymogi otwartości opartej na integracji z Internetem. Protokół TCP/IP został zaprojektowany z myślą o uniwersalnym otwartym mechanizmie przenoszenia

danych w heterogenicznym środowisku intersieciowym. Otwartość daje duże nadzieje co do jego możliwości adaptacyjnych, jednak w związku ze swoim uniwersalizmem zawiera szereg protokołów zupełnie zbędnych w typowych systemach przemysłowych. Dodatkowy problem pojawia się, gdy należy obsłużyć informację w czasie rzeczywistym. Analizując charakterystykę ruchu [9], dostępność informacji w czasie [7], sprawność oraz przepustowość [10], można stwierdzić, iż uzyskanie determinizmu czasowego zależy od obszaru pracy protokołu TCP/IP, zatem należy rozgraniczyć jego zastosowanie w zależności od rodzaju systemu, w którym ma on pracować oraz sposobu współpracy pomiędzy podsystemami. Współpraca podsystemów wykorzystujących skrajnie różne metody zarządzania dostępem do danych wymaga zastosowania obiektu pośredniczącego, wykonującego synchronizację obiegów informacji pomiędzy tymi podsystemami. Dla deterministycznych sieci systemowych oraz intersieci powinien to być *firewall* wyspecjalizowany pod kątem pracy z systemami deterministycznymi.

2. Rodzaje i współpraca podsystemów komunikacyjnych

Na rysunku 1 przedstawiono schemat informatycznego systemu przemysłowego składającego się z obiektów obsługujących proces przemysłowy oraz sieci komputerowej zapewniającej wymianę danych użytecznych pomiędzy obiektami. Obiekty te stanowią jednocześnie abonentów tej sieci.



Rys. 1. Lokalny przemysłowy system kontrolny z warstwowym podziałem zmiennych
Fig. 1. Local industrial control system with division of variables in layers manner

Przyjęto, iż jest to system lokalny wykorzystujący zmienne v_{Ai} ze zbioru zmiennych Z_A , gdzie $i = 1 \dots p$, $p < \infty$ oraz liczebność zbioru Z_A wynosi $l_A = p$. Każdy z n abonentów

obsługuje k_j zmiennych V_{Aj} , gdzie $j = 1 \dots n$. Pod względem zakresu obsługi zmiennych przez obiekty można wyodrębnić dwie grupy informacji krążących w systemie.

Pierwsza grupa informacji to zbiór zmiennych aplikacyjnych pracujących w warstwie aplikacji systemu lokalnego (Z_A):

$$Z_A = \{l_A; V_{Ap} : 0 < p \leq \infty\},$$

gdzie

l_A – liczebność zbioru Z_A .

Pomiędzy abonentami systemu lokalnego a obiektem komunikacyjnym w tym systemie zachodzi wymiana danych. Wymiany te nazwano wymianami lokalnymi. Obsługują one zmienne sieciowe V_S stanowiące drugą grupę informacji krążących w systemie (Z_S)

$$Z_S = \{l_S; V_{Sg}; 0 < g \leq l_A\},$$

gdzie

l_S – liczebność zbioru Z_S .

Pomiędzy zmiennymi aplikacyjnymi a zmiennymi sieciowymi zachodzi współzależność. Zmienne sieciowe są konstruowane ze zmiennych aplikacyjnych, gdyż celem pracy warstwy komunikacyjnej jest przekazywanie zmiennych aplikacyjnych między rozproszonymi aplikacjami systemu. Jednak nie zawsze zbiory tych zmiennych są tożsame. Dla systemu przemysłowego można stwierdzić, iż:

$$l_A = \sum_{j=1}^n l_{Aj} = const, \quad (1)$$

gdzie:

n – liczba abonentów lokalnych systemu,

oraz

$$l_S = \sum_{j=1}^n l_{Sj} = const. \quad (2)$$

Stała liczebność zbioru Z_A (wzór 1) stanowi warunek zachowania determinizmu działania systemu. Rozmiar danych wchodzących w skład zbioru Z_A jest również stały, natomiast rozmiar danych w zbiorze zmiennych sieciowych Z_S nie musi być stały, ponieważ grupa informacji przesyłanej siecią składa się z podzbioru informacji użytecznych Z_A oraz informacji pochodzących od narzutu poszczególnych warstw wykorzystywanych protokołów.

Dynamiczny charakter rozmiaru danych nie wpływa na liczebność zbioru Z_S , gdyż informacje narzutu protokołów nie stanowią nowych zmiennych systemowych a jedynie informację serwisową umożliwiającą transfer danych pomiędzy aplikacjami. Ponadto, ponieważ liczba zmiennych komunikacyjnych może być co najwyżej równa liczbie zmiennych aplikacyjnych oraz liczba ta nie zmienia się w trakcie pracy systemu, można stwierdzić, iż liczebność zbioru Z_S jest również stała (wzór 2).

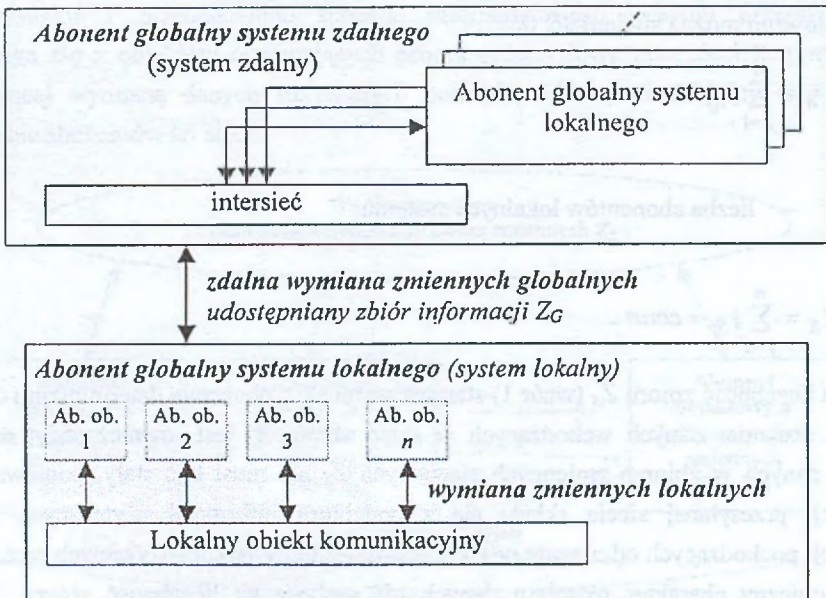
Zmienne ze zbioru Z_S stanowią zmienne lokalne lokalnego systemu komunikacyjnego i za ich pomocą realizowane są wymiany wewnętrzne w systemie lokalnym. Zmienne, które są udostępniane poza system lokalny (zbiór Z_G), stanowią zmienne globalne. Zilustrowano to na rysunku 2.

Przekazywanie informacji użytecznej z poziomu zmiennych lokalnych na poziom zmiennych globalnych i odwrotnie w systemach przemysłowych powinno się odbywać za pomocą specjalnej warstwy aplikacyjnej.

Dla uproszczenia reprezentacji podsystemów lokalnego i zdalnego abonenci tych podsystemów wraz ze swoimi systemami komunikacyjnymi reprezentowani będą przez abonentów globalnych. Sprowadzenie systemu kontrolnego do obiektów globalnych z zaznaczeniem wymian informacji między obiektami przedstawiono na rysunku 2.

Pojawia się zatem dwójaka dziedzina wykorzystywania protokołu TCP/IP: w wymianie lokalnej oraz w wymianie zdalnej.

W celu zastosowania protokołu TCP/IP w systemach przemysłowych należy usystematyzować budowę systemów przemysłowych z punktu widzenia współdziałania lokalnego oraz zdalnego abonenta globalnego. Istnieje co najmniej kilka przypadków takiej współpracy.

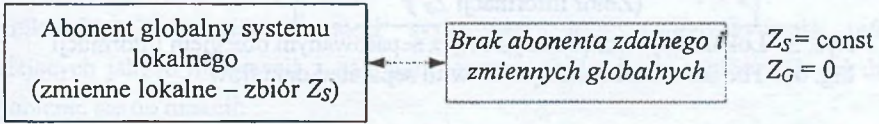


Rys. 2. Podział zmiennych i globalizacja abonentów

Fig. 2. Variables classification and globalisation of subscribers

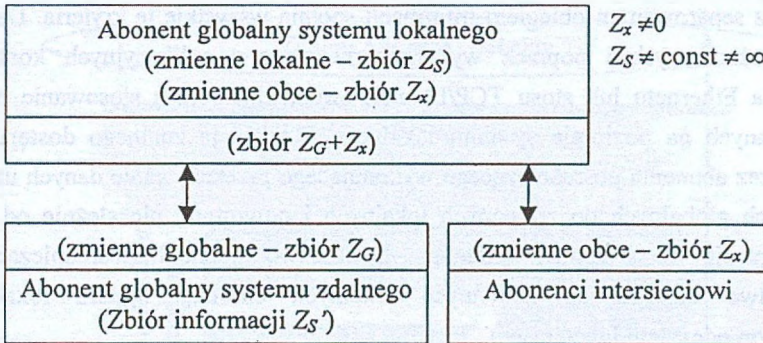
2.1. Współpraca abonentów globalnych

Jeżeli liczba zmiennych lokalnych jest ściśle określona na etapie konfiguracji systemu i nie ulega zmianie w czasie jego pracy oraz brak jest zmiennych globalnych, to sieć systemowa jest siecią zamkniętą, a system ma obieg informacji zamknięty w granicach systemu lokalnego. Czyli gdy $Z_S = \text{const}$, $Z_G = 0$, wówczas otrzymuje się lokalny system przemysłowy z zamkniętym obiegiem informacji (rys. 3).



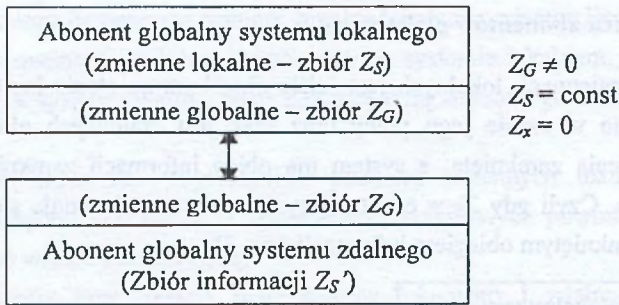
Rys. 3. Lokalny system przemysłowy z zamkniętym obiegiem informacji
 Fig. 3. The industrial local system with closed data flow

W przypadku przeciwnym, gdy w sieci systemowej krąży informacja nie związana z systemem, wówczas sieć i system będą miały charakter otwarty (rys. 4). Stanowi to przypadek pracy warstwy komunikacyjnej bezpośrednio w intersieci. Określanie liczebności zbioru Z_G jest bez znaczenia, gdyż obsługa sieciowa jest taka sama niezależnie od liczby zmiennych globalnych.



Rys. 4. Lokalny system przemysłowy z otwartym obiegiem informacji
 Fig. 4. The industrial local system with open data flow

Istnieje również trzeci przypadek, w którym sieć systemowa nie obsługuje zmiennych spoza systemu $Z_S = \text{const}$, natomiast liczba zmiennych globalnych jest niezerowa. Wówczas można mówić o systemie otwartym z kontrolowanym przepływem informacji pomiędzy systemem lokalnym a systemem zdalnym, czyli o lokalnym systemie separowanym (rys. 5).



Rys. 5. Lokalny system przemysłowy z separowanym obiegiem informacji
 Fig. 5. The industrial local system with separated data flow

2.2. Wybór rozwiązania optymalnego

Analizując omówione powyżej trzy przypadki konstrukcji systemu przemysłowego jako najlepszy do dalszych rozważań wybrano system otwarty z separowanym obiegiem informacji. Główne kryteria takiego wyboru to:

- zapewnienie determinizmu czasowego dostępu do danych w systemie lokalnym,
- zapewnienie możliwości zdalnego dostępu,
- maksymalizacja bezpieczeństwa dostępu do danych systemu lokalnego.

System z separowanym obiegiem informacji spełnia wszystkie te kryteria. Determinizm czasowy można uzyskać poprzez wykorzystanie warstw aplikacyjnych kontrolujących wymiany dla Ethernetu lub stosu TCP/IP oraz ewentualnie przez stosowanie protokołów specjalizowanych na poziomie systemu lokalnego. Realizacja zdalnego dostępu staje się możliwa przez abonenta pośredniczącego wykonującego przekazywanie danych użytecznych ze zmiennych globalnych do zmiennych lokalnych i odwrotnie, niezależnie od tego, jaki protokół pracuje w systemie lokalnym. Dodatkowo abonent pośredniczący dba o bezpieczeństwo dostępu do zmiennych lokalnych chroniąc system lokalny przed niepowołanym odczytem lub zapisem.

Zalety tak skonstruowanego systemu to przede wszystkim:

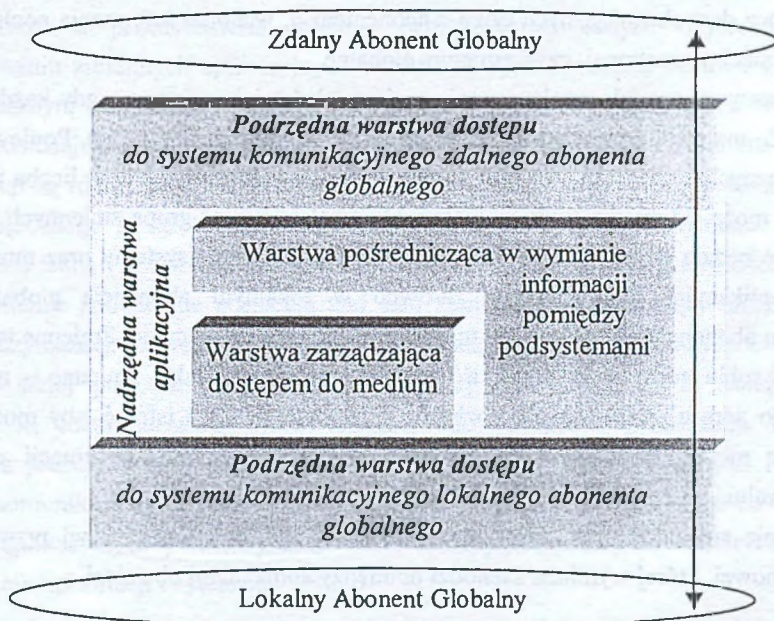
- zachowanie cech zdeterminowanego w czasie łącza komunikacyjnego systemu lokalnego,
- duża przepustowość łącza komunikacyjnego systemu lokalnego,
- atrakcyjność ekonomiczna,
- możliwość uzyskania zdalnego dostępu,
- standaryzacja zdalnego dostępu,
- możliwość standaryzacji łącza komunikacyjnego systemu lokalnego,
- możliwość integracji z określaniem jakości usług,
- wysoki poziom bezpieczeństwa zdalnego dostępu,
- brak wpływu pracy abonentów globalnych na siebie.

3. Budowa warstwy aplikacyjnej

Budowa sprzętu (zastosowane koprocesory, konstrukcje pamięci, magistral itp.) abonenta pośredniczącego w wymianie zmiennych między abonentami globalnymi nie wpływa na logiczną konstrukcję stosu protokołu. Kluczowy problem stanowi budowa warstwy aplikacyjnej, stanowiącej nadrzędną warstwę spełniającą funkcje filtracji i przekazywania pakietów, buforowania i synchronizacji obiegów danych związanych z interfejsami komunikacyjnymi intersieci i sieci systemowej oraz wypracowywania informacji określających jakość informacji z poziomu warstwy aplikacji abonentów [5]. Jej działanie musi opierać się na funkcji:

- 1) kontrolującej dostęp do medium lokalnej sieci systemowej,
- 2) realizującej zdalny dostęp do systemu lokalnego.

W celu zapewnienia determinizmu czasowego w systemie lokalnym warstwa kontrolująca dostęp musi być zaimplementowana w interfejsie każdego abonenta tego systemu. Warstwa realizująca zdalny dostęp jest niezbędna tylko dla abonentów pośredniczących w wymianie informacji użytecznej pomiędzy systemem lokalnym a systemem zdalnym. Zostało to przedstawione na rysunku 6.



Rys. 6. Budowa warstw deterministycznego interfejsu komunikacyjnego abonenta sieci Ethernet

Fig. 6. Construction of deterministic layers of the Ethernet subscriber's communication interface

Warstwa zarządzająca musi pracować realizując scenariusz wymian zgodnie z jednym z deterministycznych modeli wymian. Jej działanie jest pochodną działania tych modeli. Na potrzeby testów wykonano nadbudowę na bazie modelu PDC [6, 8].

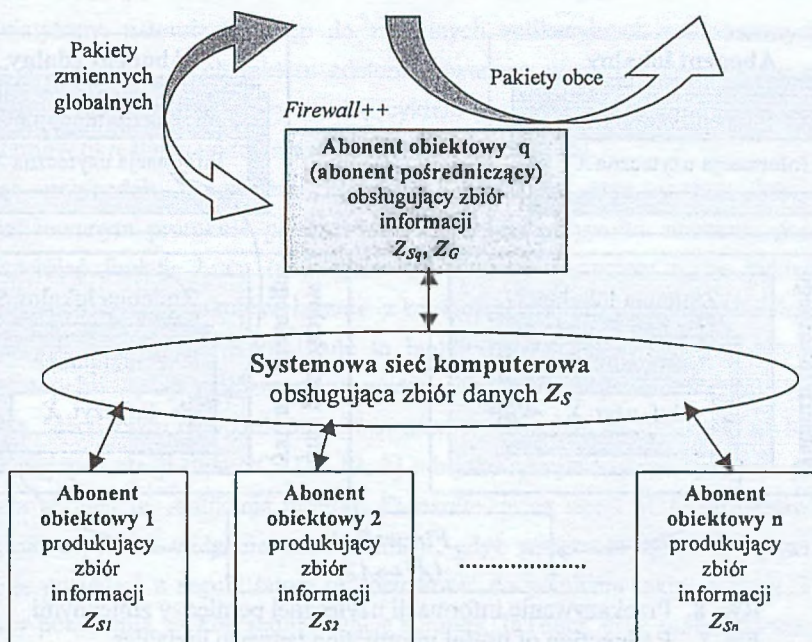
Warstwa separująca pod względem realizowanych funkcji przypomina działanie tzw. ściany ogniowej (ang. *firewall*). Ideę tę przedstawiono na rysunku 7. Istnieją komercyjne rozwiązania pakietów typu „*firewall*” dostarczane dla systemów przemysłowych. Przykładem może być urządzenie Firewall FWA-230 firmy Advantech, gdzie na specjalizowanym komputerze pracuje system operacyjny Linux RedHat oraz oprogramowanie Check Point™ FireWall-1® lub Check Point™ VPN-1®. Są to jednak urządzenia, w których pakiety kontrolujące dostęp działają jak pakiety standardowe bez dodatkowych funkcji uwzględniających specyfikę ruchu w sieciach przemysłowych.

Poza klasyczną funkcją filtrowania pakietów warstwa ta musi wykonywać dodatkowe funkcje, mające na celu wprowadzenie zmiennych systemu komunikacyjnego zewnętrznego do systemu wewnętrznego bez zakłócania tego drugiego. Proponuje się nazywać abonenta posiadającego taką warstwę separującą terminem *Firewall++*.

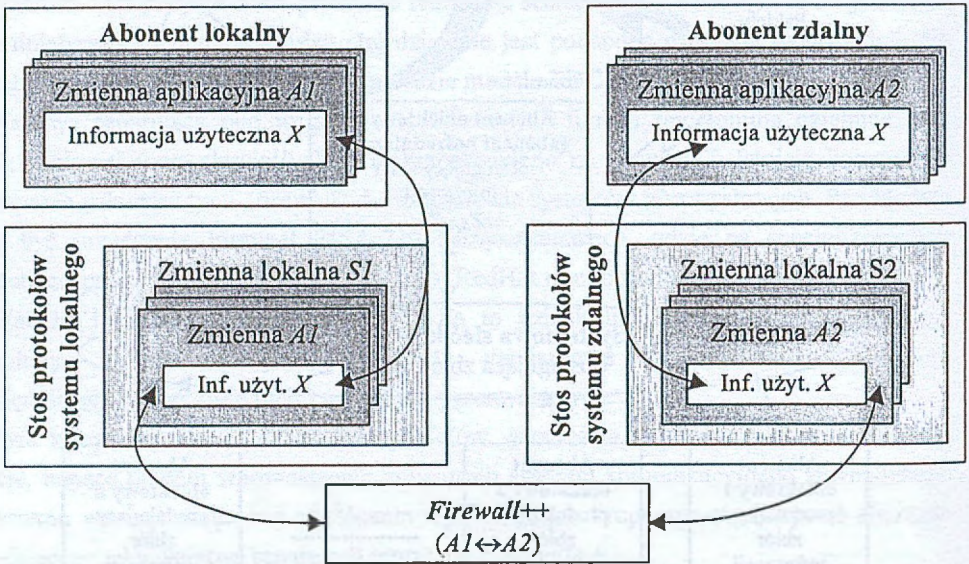
Na rysunku 7 zbiór zmiennych Z_{Sg} jest obsługiwany przez system komunikacyjny zgodnie z zasadami pracy sieci wewnętrznej i z punktu widzenia systemu lokalnego są to informacje produkowane i konsumowane przez abonenta g . W rzeczywistości w skład tej grupy, oprócz danych związanych tylko z abonentem g , wchodzi informacja pochodząca od abonentów sieci zewnętrznej, czyli zmienne globalne.

Najprostszy przypadek przekazywania zmiennych istnieje wówczas, gdy każda zmienna ze zbioru Z_G ma swój odpowiednik w zbiorze Z_{Sg} i odwrotnie, ($Z_G = Z_{Sg}$). Ponieważ jednak liczba fizycznych abonentów systemu zdalnego nie jest określona, zatem liczba zmiennych globalnych może się dynamicznie zmieniać. Musi zatem istnieć grupa zmiennych lokalnych, która służyć będzie do przesyłania informacji użytecznej spoza systemu oraz muszą istnieć zmienne aplikacyjne funkcjonujące zarówno w lokalnym abonencie globalnym, jak i w zdalnym abonencie globalnym. Są to globalne zmienne aplikacyjne. Zmienne takie muszą zawierać w sobie adres aplikacyjny, a abonenci obsługujący takie zmienne – mechanizm obsługi tego adresu. Adresacja na poziomie aplikacyjnym musi istnieć, aby możliwe było przekazanie między abonentami lokalnymi i zdalnymi dowolnej informacji użytecznej, identyfikowalnej na poziomie warstw aplikacji.

Proponuje się zatem przekazywanie systemowej informacji użytecznej przez protokół sieci systemowej, której wymiana zachodzi pomiędzy abonentami obu sieci.

Rys. 7. System lokalny z abonentem *Firewall++*Fig. 7. The local system with a *Firewall++* subscriber

Na rysunku 8 przedstawiono schemat przekazywania danych użytecznych przy wykorzystywaniu zmiennych aplikacyjnych oraz sieciowych. Zmienne sieciowe $S1$ i $S2$ są różne pod każdym względem. Zawierają inny zestaw zmiennych aplikacyjnych, inne adresy sieciowe, posiadają różny rozmiar. Zmienne aplikacyjne $A1$ i $A2$ z punktu widzenia implementacji są różne, gdyż różna może być platforma implementacyjna. Jednak z punktu widzenia logicznego są to te same zmienne, stanowiące aplikacyjne zmienne globalne. Obsługują one jedną i tę samą informację użyteczną X . W skali całego systemu kontrolnego globalne zmienne aplikacyjne posiadają taki sam identyfikator, stanowiący unikalny adres informacji użytecznej. Zadanie preadresowania spoczywające na *Firewall++* polega na skojarzeniu takiej pary zmiennych sieciowych, aby obie zawierały tę samą zmienną aplikacyjną A , identyfikowaną przez unikalny identyfikator i przepisaniu informacji użytecznej z jednej do drugiej zmiennej sieciowej S . Przy użyciu lokalnych i zdalnych systemów komunikacyjnych oraz *Firewall++* następuje stworzenie tunelu do transmisji zmiennych aplikacyjnych pomiędzy abonentami lokalnymi i zdalnymi, w którym zachodzi przekazywanie informacji użytecznej.



Rys. 8. Przekazywanie informacji użytecznej pomiędzy zmiennymi
Fig. 8. Redirection of useful information between variables

Cechę charakterystyczną takiego tunelu stanowi zapewnienie zdeterminowanego w czasie dostępu do zmiennych sieciowych w tej jego części, która przebiega przez system lokalny, pomimo jednoczesnej realizacji niezdedeterminowanego w czasie przesyłu globalnych zmiennych aplikacyjnych.

3.1. Przypadki zestawiania warstw podrzędnych

Przy separacji systemu lokalnego i zdalnego można stosować szereg kombinacji protokołów z wykorzystaniem stosu TCP/IP.

Pierwszy charakterystyczny przykład stanowi kombinacja systemu zdalnego na bazie sieci Internet z protokołem TCP/IP oraz systemu lokalnego opartego na standardzie Ethernet również z wykorzystaniem stosu TCP/IP. Przypadek ten jest przypadkiem najprostszym. Odfiltrowanie pakietów IP polega na odrzuceniu pakietów, których adresy nie odnoszą się do interfejsów sieci systemowej. Wystarczy zastosować klasyczne mechanizmy filtrowania stosowane w urządzeniach typu ściana ogniowa. Przeadresowanie zmiennych dla przypadku, gdy $Z_G = Z_{S_g}$, polega na zamianie adresacji IP widzianej na zewnątrz na adresację systemu lokalnego i odwrotnie. W sytuacji gdy równość między zbiorami nie zachodzi, należy dodatkowo przeadresowywać zmienne aplikacyjne. System może pracować zarówno z wykorzystaniem mechanizmów determinizmu czasowego, jak i bez niego. W przypadku stosowania kontroli wymian niezbędne staje się zaimplementowanie na stacji pośredniczącej mechanizmu buforowania pakietów. Dostęp do zmiennych lokalnych jest wówczas

deterministyczny, natomiast dostęp do zmiennych aplikacyjnych przenoszonych przez te zmienne lokalne nie ma charakteru zdeterminowanego w czasie. Pojawia się niespójność czasowa pomiędzy cyklem pracy sieci a cyklem pracy aplikacji. Wymusza to stosowanie mechanizmów określania takich niespójności, opisanych w [5].

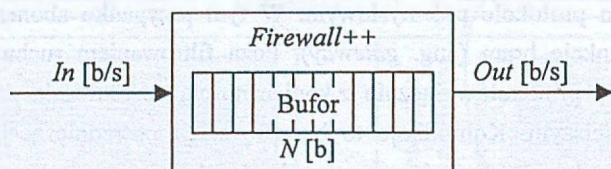
Drugi przypadek to system zdalny z intersiecią oraz system lokalny oparty na specjalizowanym protokole przemysłowym. W tym przypadku abonenci pośredniczący muszą spełniać funkcję bram (ang. *gateway*). Poza filtrowaniem ruchu pakietów muszą wykonać konwersję protokołów łącznie z koniecznością adresowania danych, tak jak w przypadku wcześniejszym. Komplikuje to budowę stacji pośredniczącej, lecz zyskujemy wówczas w pełni wydajny przemysłowy system komunikacyjny z możliwością współpracy z intersiecią. Mechanizm pośredniczący *Firewalla++* może być zrealizowany na bazie warstw komunikacyjnych stacji typu SCADA [2, 3] lub jako specjalizowana aplikacja pracująca na wybranym abonencie. Aplikacja modułu *Firewalla++* na stacji SCADA stanowi wygodne rozwiązanie z punktu widzenia implementacji, gdyż większość systemów przemysłowych taką stację posiada i z reguły łatwo ją dostosować do pełnienia takiej funkcji. W praktyce wykonano połączenie wykorzystując stację Kronos, InTouch oraz moduł *Firewalla++* dla sieci WorldFip oraz testowego protokołu UDP/PDC.

Kolejne rozwiązanie stanowi wykorzystanie protokołu TCP/IP do połączenia dwóch systemów pracujących na bazie protokołów przemysłowych poprzez intersieć. Najprostszym rozwiązaniem tego problemu będzie tunelowanie ramek protokołów przemysłowych w kanale transmisji IP. Problemy determinizmu w dostępie do wartości tunelowanych zmiennych wówczas nie zanikają i należy je rozwiązywać na poziomie warstw aplikacji.

3.2. Określenie jakości usług przekazywania danych

Ponieważ podstawowe zadanie *Firewalla++* to przekazywanie danych z jednego do drugiego podsystemu komunikacyjnego, należy charakteryzując *Firewalla++* określić jakość takiej usługi. Cykle sieci systemowej oraz intersieci nie są zsynchronizowane, tak samo jak obiegi informacji w interfejsie komunikacyjnym *Firewalla++*. Może zatem zaistnieć sytuacja, iż strumień danych przychodzących do abonenta pośredniczącego nie będzie równy strumieniowi wyjściowemu. Wynika to z faktu, że cykl zdeterminowany sieci systemowej generuje strumień stały lub określony w przedziale oraz umożliwia wprowadzenie do obiegu liczbę danych określoną przez wartość maksymalną. Natomiast cykl intersieci jest nieokreślony, więc nieprzewidywalny. Jednocześnie nie wolno dopuścić do utraty danych. *Firewall++* nie gwarantuje spójności czasowej przekazywanych danych, lecz jednocześnie nie powinien podczas poprawnej pracy powodować utraty przekazywanych zmiennych, a także zmieniać kolejności przekazywania zmiennych.

Aby zagwarantować jakość usług (ang. *quality of service*) przekazywania zmiennych takiego oprogramowania, należy zapewnić wspomniane wcześniej buforowanie nadchodzących pakietów. W sytuacji gdy strumień wejściowy będzie większy od wyjściowego, *Firewall++* powinien być w stanie zapewnić przekazywanie danych przy określonym strumieniu przez określony czas.



Rys. 9. Buforowanie danych
Fig. 9. Buffering of data

Stosując mechanizm buforowania zmiennych i zakładając, że czas obsługi danych jest mniejszy od minimalnego periodu cyklu sieci docelowej, *Firewall++* jest w stanie zapewnić jakość usług przekazywania danych na poziomie:

$$Q = \frac{N}{In - Out} [s].$$

Oznacza to, że warstwy oprogramowania pośredniczącego umożliwiają przekazywanie danych bez ich utraty przez Q sekund przy strumieniu wejściowym In [b/s] i wyjściowym Out [b/s]. Zagadnienie jakości obsługi w *firewallu++* można również poddać analizie kolejkowej traktując go jako stanowisko obsługi, gdzie intersieciowy strumień wejściowy/wyjściowy jest opisany rozkładem prawdopodobieństwa [4].

3.3. Podsumowanie cech *Firewalla++*

Systemy oparte na *Firewallu++* mogą służyć do realizacji aplikacji przemysłowych wymagających ograniczeń czasowych, jednak dostęp nie do wszystkich danych użytecznych obsługiwanych przez ten system ma charakter deterministyczny. *Firewall++* nie gwarantuje spójności czasowej aplikacyjnych zmiennych globalnych, lecz może zapewnić obsługę mechanizmów określania tej niespójności oraz gwarancję przekazania każdej kolejnej zmiennej przez określony czas. Istnieją zatem ograniczenia w tworzeniu aplikacji wymagających determinizmu, lecz dla wybranej grupy informacji możliwe jest tworzenie aplikacji czasu rzeczywistego. Dzięki idei *Firewalla++* można skonstruować jednoczesną obsługę niezdeteminowanych w czasie zmiennych globalnych w zdeterminowanym cyklu obsługi zmiennych lokalnych.

LITERATURA

1. Comer D.: Sieci komputerowe i intersieci. WNT, Warszawa 2001.
2. Cupek R.: Wizualizacja rozproszonych procesów przemysłowych. Rozprawa doktorska, Instytut Informatyki Politechniki Śląskiej, Gliwice 1998.
3. Cupek R.: Protokół TCP/IP w systemach wizualizacji procesów przemysłowych. ZN Pol. Śl. s. Studia Informatica Vol. 22 No 3, Gliwice 2001.
4. Czachórski T.: Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych. WPKJS, Gliwice 1999.
5. Gaj P.: Określanie jakości informacji użytecznej przy stosowaniu protokołu TCP/IP w informatycznych systemach przemysłowych. Materiały Konferencyjne SCR'02, Gliwice 2002.
6. Gaj P.: Szybka sieć przemysłowa a system wizualizacji – problem interfejsu. ZN Pol. Śl. s. Informatyka z. 36, Gliwice 1999.
7. Kwiecień A., Bigewski Z., Mrówka Z.: Analiza czasu najgorszego przypadku w sieciach przemysłowych.. ZN Pol. Śl. s. Informatyka z.36, Gliwice 1999.
8. Kwiecień A., Gaj P., Grzywak A., Mrówka Z.: Rozwiązania sprzętowe i programowe sieci przemysłowej FIP. ZN Pol. Śl. s. Informatyka z. 30, Gliwice 1996.
9. Kwiecień A., Gaj P.: Dobór protokołów w sieciach przemysłowych. ZN Pol. Śl. s. Studia Informatica Vol. 22 No 3, Gliwice 2001.
10. Kwiecień A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych. ZN Pol. Śl. s. Studia Informatica Vol. 23, No 1, Gliwice 2002.

Recenzent: Dr inż. Włodzimierz Boroń

Wpłynęło do Redakcji 1 kwietnia 2003 r.

Abstract

Cooperation between subsystems using different methods of data access requires using of intermediate object that could synchronise the data flow between these subsystems. For deterministic field networks and internetworks there should be a *firewall* specialised from the point of view of deterministic systems work. *Firewall* application layer operation must be based upon following functions:

1. intermediate access control of local field network,
2. remote access to local system.

A management layer must work as a realization of exchanges scenario according to one of the deterministic exchanges' models. The work idea of separating layers is shown in the picture 7. Apart from filtering function of classical packets this layer must realize an additional function with the aim of putting external communication variables into deterministic internal system without disturbing a letter.

Adresy

Piotr GAJ: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-101 Gliwice, Polska, pgaj@top.iinf.polsl.gliwice.pl.

Józef OBER: Instytut Informatyki Teoretycznej i Stosowanej PAN, ul. Bałtycka 5, 44-100 Gliwice, Polska, jober@top.iinf.polsl.gliwice.pl.