

Piotr KASPRZYK, Mirosław SKRZEWSKI, Adam DOMAŃSKI
Politechnika Śląska, Instytut Informatyki

MECHANIZMY GENERACJI ZBĘDNEGO RUCHU W SIECI

Streszczenie. Poniższy artykuł przedstawia problem pojawiający się w sieciach o złożonej topologii. Na skutek powielania się ramek rozgłoszeniowych powstaje zjawisko burzy broadcastowej. Duże liczby ramek krążących w sieci powodują załamanie jej pracy. W artykule przedstawiono model matematyczny problemu i metody unikania takich sytuacji.

THE GENERATION OF NEEDLESS NETWORK DATA

Summary. The article discusses issue existing in complex topology LAN. Broadcast frames duplicate and cause broadcast storm. Huge quantities of frames are transmitted in the net, stopping its normal way of working. The article present mathematical model of this problem and methods of avoiding this situation.

1. Wstęp

Od pewnego czasu obserwowano pogorszenie efektywności działania sieci lokalnej Zakładu, objawiające się występowaniem długich czasów reakcji w programach uruchamianych na serwerach sieci lokalnej, sięgających niekiedy do kilku sekund, oraz małymi i bardzo małymi szybkościami transmisji przy korzystaniu z Internetu. Sporadycznie obserwowano również występowanie błędów braku wolnych bloków ECB na serwerze *TOP*, prowadzące niekiedy do „zawieszenia” się serwera na okres do kilkudziesięciu minut. Przypadkowo, w trakcie zajęć laboratoryjnych poświęconych monitorowaniu ruchu pakietów w sieci prowadzonych w sobotę w godzinach południowych, przy praktycznie nieobciążonej sieci zaobserwowano pojawianie się okresowo dużych skoków aktywności sieci (występowania w okresie 2 – 3 sekund nawet do 10 – 20 tysięcy ramek), w których ponad 95 % stanowiły broadcasty (transmisje rozgłoszeniowe).

Prowadzone w ciągu kolejnych następných dni obserwacje potwierdziły występowanie wyraźnych impulsowych skoków aktywności sieci, a gromadzone statystyki pokazywały przy normalnej pracy sieci udział broadcastów przekraczający 80 %. Działające na komputerach oprogramowanie w żaden sposób nie uzasadniało tak dużego udziału transmisji rozgłoszeniowych w pracy sieci. Ponieważ na podobnych zajęciach prowadzonych w poprzednich semestrach udział ramek rozgłoszeniowych nie przekraczał kilku (2-3) procent, tak duży wzrost liczby ramek typu broadcast należało wiązać z zainstalowanym w ciągu ostatniego roku w sieci nowym sprzętem, oprogramowaniem lub dokonanymi zmianami w konfiguracji lub topologii sieci. W celu wskazania potencjalnych czynników mogących odpowiadać za niekorzystne zmiany w pracy sieci przeanalizowano sposób korzystania z sieci przez zainstalowane oprogramowanie oraz obecną topologię sieci komputerowej.

2. Oprogramowanie i topologia sieci

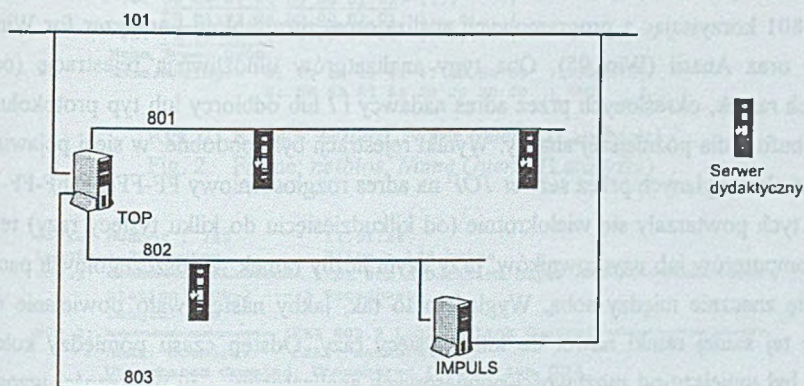
Sieć lokalna Zakładu oparta jest na systemie Netware zainstalowanym na serwerze *TOP*. W ostatnim roku wersja systemu 3.11 zastąpiona została nowym systemem 4.1 na większą liczbę użytkowników, dokonano również wymiany komputera serwera na szybszy, z większą pamięcią RAM i zwiększoną pojemnością pamięci dyskowej.

Oprogramowanie serwera wykorzystuje głównie protokół IPX (NCP) pracujący w trybie datagramowym, protokoły pomocnicze RIP, SAP oraz sporadycznie SPX. Z powyższych protokołów z transmisji rozgłoszeniowych (broadcastów) korzystają tylko protokoły RIP i SAP do rozgłaszania aktualnych tablic routingu oraz listy dostępnych w sieci usług, wysyłając cyklicznie co 60 sekund po kilka ramek rozgłoszeniowych. W wersji Netware 4.1 protokoły te zostały zastąpione protokołem NLSP (Netware Link Service Protocol), który nie rozsyła tak często jak jego poprzednicy ramek typu broadcast, robi to rzadziej i ogranicza się do 2 - 3 ramek raz na kilka minut.

Aplikacje internetowe, oprogramowanie Windows 95, NT, Unix wykorzystują inną rodzinę protokołów, TCP/IP. W rodzinie tej z transmisji rozgłoszeniowych korzystają protokoły pomocnicze ARP (Address Resolution Protocol) i RARP, w trakcie ustalania powiązań pomiędzy adresem sieciowym IP a numerem karty sieciowej komputera (adresem warstwy liniowej). Oprogramowanie systemów Windows wykorzystuje również protokół NetBIOS oraz jego firmową wersję NetBEUI, posługujące się w niektórych fazach działania transmisjami rozgłoszeniowymi. Oprogramowanie routera wbudowanego w serwer Netware nie przepuszcza ramek rozgłoszeniowych do sąsiednich segmentów sieci (izoluje broadcasty do segmentu, w którym zostały wygenerowane). Jedyne dla ramek NetBIOSa czyniony jest

wyjątek. Korzystają one ze specjalnego typu pakietu protokołu IPX, tzw. *NetBIOS Name Propagated Packet* (identyfikator 14h), który przepuszczany jest przez routery do sąsiednich segmentów sieci, z silniejszym ograniczeniem czasu życia pakietów do 8 węzłów topologii.

Konfiguracja sieci lokalnej Zakładu oparta jest na trzech podsieciach obsługiwanych przez serwer *TOP*, połączony czwartą kartą sieciową do sieci szkieletowej wydziału (backbone). Laboratoria i pomieszczenia pracowników IX p. połączone są podsiecią 801, laboratoria VIII piętra obsługuje podsieć 802, a pozostałe pomieszczenia Zakładu wraz z sekretariatem Instytutu połączone są podsiecią 803. Sieć szkieletowa wydziału ma nadany numer 101. Konfigurację sieci przedstawia rys. 1.



Rys. 1. Konfiguracja sieci lokalnej Zakładu
Fig. 1. Configuration of Department's LAN

Numeracja segmentów podsieci obsługiwanych przez serwery Netware ma swoją specyfikę związaną z budową oprogramowania serwera. Numer identyfikujący sieci nadawany jest nie łączu fizycznemu danej podsieci, lecz kanałowi transmisji obsługiwanemu przez zainstalowany protokół warstwy liniowej (żargonowo „zabindowany” do karty sieciowej) danego łącza fizycznego w serwerze. Jeśli w danym łączu fizycznym obsługiwane są dwa lub więcej protokoły liniowe (np. 802.3, 802.2, Ethernet II), to każdy z protokołów otrzymuje niezależny identyfikator dla swojego (tzn. utworzonego przy jego pomocy) kanału transmisji. Przykładowo, na serwerze *TOP* podsieć IX piętra może występować odpowiednio pod numerami 801, 804 lub 807.

Oprócz serwera *TOP* w sieci Zakładu zainstalowanych jest na stałe kilka innych serwerów (*IMPULS*, *IMPULS_OLD* – Netware 3.11, *Maleństwo* – LINUX, Winserv – NT, - wszystkie na podsieci 802) oraz zmienna w czasie liczba (2 – 3) serwerów laboratoryjnych uruchamianych w trakcie zajęć dydaktycznych w podsieciach 801 i 802. Od pewnego czasu

jeden z serwerów (*IMPULS*) połączony jest niezależnie od *TOP-a* z siecią szkieletową wydziału (101), tworząc alternatywną drogę (pętlę) łączącą podsieć 802 z siecią zewnętrzną (backbone).

3. Wyniki rejestracji ruchu ramek w sieci

Ponieważ przeprowadzona analiza konfiguracji nie dostarczyła wskazówek co do przyczyny występowania tak dużej liczby broadcastów w sieci, zdecydowano się na dokładną analizę treści przesyłanych siecią ramek rozgłoszeniowych. Rejestrację prowadzono na podsieci 801 korzystając z programowych analizatorów protokołów Lanalyzer for Windows (Win3.x) oraz Anasil (Win 95). Oba typy analizatorów umożliwiają rejestrację (odbiór) wybranych ramek, określonych przez adres nadawcy i / lub odbiorcy lub typ protokołu, i ich zapis do bufora dla późniejszej analizy. Wyniki rejestracji były podobne: w sieci pojawiały się paczki ramek wysyłanych przez serwer *TOP* na adres rozgłoszeniowy FF-FF-FF-FF-FF-FF, w ramach tych powtarzały się wielokrotnie (od kilkudziesięciu do kilku tysięcy razy) te same nazwy komputerów lub użytkowników, przy czym liczby ramek w poszczególnych paczkach różniły się znacznie między sobą. Wyglądało to tak, jakby następowało powielanie niemal dokładne tej samej ramki nawet do kilku tysięcy razy. Odstęp czasu pomiędzy kolejnymi ramkami był mniejszy od możliwości pomiarowych analizatorów – w większości wypadków podawały różnicę czasu 0 ms. Tego typu zjawisko określano w literaturze jako „burze broadcast-owe”. Zarejestrowane ramki reprezentowały kilka typów protokołów, przy czym powtarzały się głównie dwie wersje ramek NetBIOS'a, przedstawione na rys. 2 i 3, różnie raportowane przez analizator protokołów, raz jako ramka protokołu *Novell Netbios Name Query* (rys. 2), drugi raz jako ramka protokołu *IPX, typu Netbios Name* (rys.3). Różnice między nimi sprowadzały się do różnych numerów gniazdek nadawcy i odbiorcy. Protokół *Novell NetBIOS* posługuje się gniazdkiem 0455, natomiast druga wersja (*IPX, Netbios Name*) korzystała z różnych gniazdek - 0550, 0551, 0553.

W podobny sposób treść zarejestrowanych ramek, choć mniej dokładnie, przedstawiał analizator Anasil (rys. 4).

Zawarte w ramach informacje w zasadzie pokrywały się, choć były różnie przedstawiane przez program, i składały się z informacji o przebytej trasie (*Router Information*) oraz rozgłaszanej nazwy z pomocniczymi parametrami. Pole *Router Information* pozwala na zarejestrowanie do 8 numerów podsieci, pokonanych przez daną ramkę (numery podsieci podkreślono).

```

Packet Number : 3422          10:22:21
Length : 98 bytes
802.3: ----- IEEE 802.3 Datalink Layer -----
Station: TOP-2-2 ----> Broadcast
Length: 80
ipx: ----- Internetwork Packet Exchange -----
Checksum: 0xFFFF
Length: 80
Hop Count: 7
Packet Type: 20 (NetBIOS Name Packet)
Network: 00 01 03 29 ----> 00 00 08 01
Node: 00-60-97-83-46-02 ----> FF-FF-FF-FF-FF-FF
Socket: NetBios ----> NetBios
nbios: ----- Novell NetBIOS -----
Packet Type: 1 (Name Query)
Router Information:
  0: 00 01 03 29 00 01 03 27 |...)|
  8: 00 00 01 00 00 00 01 02 |.....|
 10: 00 01 03 26 00 00 01 01 |...&...|
 18: 00 00 08 02 00 00 00 00 |.....|
Name Type: 0x00
Name String:  0: 45 4E 45 52 47 45 54 59 |ENERGETY|
              8: 5F 53 52 56 20 20 20 20 |_SRV |

```

Rys. 2. Ramka: *netbios, Name Query* (Lanalyzer)Fig. 2. Frame: *netbios, Name Query* (Lanalyzer)

```

Packet Number : 745          14:51:34
Length : 238 bytes          Slice : 128 bytes
802.3: ----- IEEE 802.3 Datalink Layer -----
Station: TOP-2 ----> Broadcast
Length: 220
802.2: ----- IEEE 802.2 Logical Link Control -----
SSAP: NetWare      DSAP: NetWare
Unnumbered Command: Unnumbered Information (UI)
ipx: ----- Internetwork Packet Exchange -----
Checksum: 0xFFFF
Length: 217
Hop Count: 2
Packet Type: 20 (NetBIOS Name Packet)
Network: 00 00 00 01 ----> 00 00 08 04
Node: 00-00-00-00-00-01 ----> FF-FF-FF-FF-FF-FF
Socket: 0x0553 ----> 0x0553
Data:
  0: 00 00 01 02 00 00 01 01 00 00 00 00 00 00 00 00 |.....|
 10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
 20: FC 03 00 00 5A 4D 49 54 41 43 20 20 20 20 20 20 |...ZMITAC|
 30: 20 20 20 00 42 4F 59 20 20 20 20 20 20 20 20 |...BOY|
 40: 20 20 20 00 FF 53 4D 42 25 00 00 00 00 00 00 00 |...SMBt.....|
 50: 00

```

Rys. 3. Ramka: *ipx, type = Netbios Name* (Lanalyzer)Fig. 3. Frame: *ipx, type = Netbios Name* (Lanalyzer)

Dokładniejsza analiza tego pola pozwoliła na określenie prawdopodobnej przyczyny występowania tak dużej liczby tego typu ramek. Przedstawione na rys. 2 i 4 ramki mają czas życia (pole *Hop Count*) równy 7, tzn. przebyły już siedem węzłów (routerów) związanych z sieciami, których numery zapisane zostały w polu *Router Information*. Dla ramki z rys. 2 były to sieci: 10329, 10327, 100, 102, 10326, 101 oraz 802; dla ramki z rys. 4: 10926, 100, 10327, 102, 10326, 101 i 802.

```

Source: 00-60-08-18-60-21      Destination: 00-60-08-18-60-21      /Anas11 /
Frame received at: 11.92 s,    Frame size: 94

--- Layer: Ethernet 802.3
      Destination address: FF-FF-FF-FF-FF-FF
      Source address:      00-60-08-18-60-21
      Length               80

--- Layer: IPX packet header
      Checksum             FFFF
      Packet length       80
      Transport control    07
      Packet type          NetBIOS
      Dest. network        00-00-08-01
      Dest. node           FF-FF-FF-FF-FF-FF
      Dest. socket         Novell NetBIOS
      Source network       00-01-09-26
      Source node          00-20-AF-52-66-AF
      Source socket        Novell NetBIOS

--- UNDECODED
00 01 09 26 00 00 01 00 00 01 03 27 00 00 01 02  . . . . .
00 01 03 26 00 00 01 01 00 00 08 02 00 00 00 00  . . . . .
40 03 47 49 53 20 20 20 20 20 20 20 20 20 20 20  e . GIS
20 00

```

Rys. 4. Ramka protokołu *NetBIOS* zarejestrowana programem Anas11

Fig. 4. Frame of NetBIOS protocol caught by Anas11

Podsieci zaczynające się od 109xx, 103xx były podsieciami Instytutu Automatyki, a numery 100, 101, 102 były numerami sieci szkieletowej Wydziału, a dokładniej numerami kanałów transmisji tworzonych przez protokoły 802.3, 802.2 i Ethernet II.

Obecność ich wszystkich na liście nasuwała podejrzenie, że ramki *NetBIOSa* wędrowały kilkakrotnie tym samym kanałem fizycznym (segmentem kabla), za każdym razem wykorzystując inny protokół warstwy łącza logicznego (numery podsieci nigdy się nie powtarzały), co wskazywało na to, że

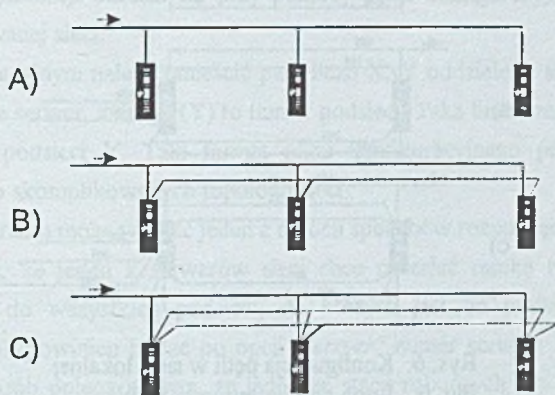
„zabindowanie” do jednego segmentu sieci (jednego kabla / karty sieciowej) więcej niż jednego protokołu może być przyczyną nadmiernego ruchu ramek rozgłoszeniowych.

Ponieważ na listach *router information* powtarzały się często numery podsieci 103xx, 104xx, 109xx, zdecydowano się zebrać dokładniejsze dane o topologii sieci w innych Instytutach Wydziału, z drugiej strony spróbować oszacować ilościowe możliwości mechanizmu powielania ramek wynikającego z wykorzystywania więcej niż jednego protokołu w segmencie sieci.

4. Model ilościowy mechanizmu powielania ramek

Rozpatrzono dwa podstawowe warianty konfiguracji sieci, które wydawały się typowe dla rozwiązań spotykanych w sieci wydziałowej:

konfiguracja liniowa – kilka serwerów podłączonych do jednego segmentu sieci (rys. 5),
 konfiguracja pętli – dwa serwery połączone są ze sobą dwoma fizycznymi segmentami,
 (rys. 6).



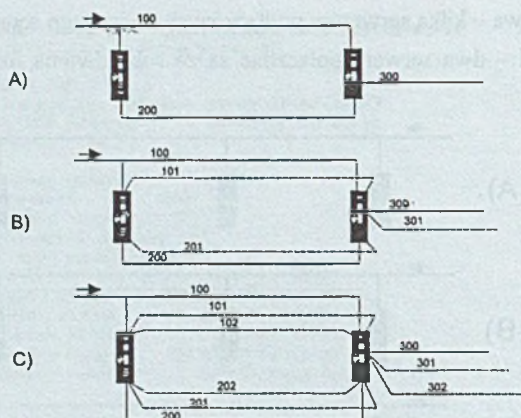
Rys. 5. Konfiguracja liniowa łączy sieci lokalnej
 Fig. 5. Bus topology of LAN

Można było rozważać dwa przypadki: do danej podsieci dociera z zewnątrz pojedyncza ramka *NetBIOSa* i zależnie od liczby zainstalowanych protokołów na segment i liczby serwerów w podsieci określamy liczbę wygenerowanych ramek oraz sytuację, w której w danej podsieci generowana jest pierwotna ramka *NetBIOSa*.

Dla sytuacji z rys. 5 założymy, że do podsieci dociera pojedyncza ramka. Przy jednym zainstalowanym protokole liniowym (rys. 5a) żaden z serwerów nie generuje ramek. Przy dwóch protokołach (rys. 5b) każdy z serwerów wysyła odebraną ramkę w drugą z dostępnych podsieci, powstaje więc N dodatkowych ramek, gdzie N to liczba serwerów. Przy trzech (lub ogólnie M) protokołach (rys. 5c) mamy dwa $(M-1)$ etapy: w pierwszym każdy z serwerów wysyła odebraną ramkę w pozostałe $(M-1)$ podsieci, powstaje więc $N*(M-1)$ dodatkowych ramek. W kolejnym etapie każda z nich w każdym z serwerów wysyłana jest w następną $(M-2)$, tzn. w ostatnią jeszcze nie wykorzystaną podsieć. Powstaje więc w tym etapie $N*(M-2) * N*(M-1)$ nowych ramek.

Przy trzech protokołach jest to ostatni etap „powielania” ramek, każda z nich ma już zarejestrowane wszystkie dostępne podsieci. Otrzymujemy więc:

- | | | |
|--------------------|---------------------------------|--------|
| - dla 1 protokołu | $1 + 0$ | ramek, |
| - dla 2 protokołów | $1 + N*1$ | ramek, |
| - dla 3 protokołów | $1 + N*(3-1) + N*(3-1)*N*(3-2)$ | ramek, |



Rys. 6. Konfiguracja pętli w sieci lokalnej
Fig. 6. Loop configuration in LAN

i ogólnie, dla N serwerów i M protokołów:

$$R = 1 + \sum_{i=1}^{M-1} \frac{(M-1)!}{(M-i-1)!} * N^i \quad \text{dla } M > 1. \quad (1)$$

Jeśli jest zainstalowany dokładnie jeden protokół dla każdego segmentu, to ramka rozgłoszeniowa pojawia się jeden raz w każdym segmencie i dlatego ten przypadek nie został uwzględniony we wzorach (wprowadzono założenie $M > 1$).

Dla przypadku przedstawionego na rys. 6 w podobny sposób otrzymujemy, że serwer A wysłał do swego otoczenia, przy M protokołach i K kartach sieciowych (segmentach łączących)

$$R_A = 1 + \sum_{i=1}^{M \cdot K - 1} \frac{(M \cdot K - 1)!}{(M \cdot K - i - 1)!} \quad \text{dla } M * K > 1, \quad (2)$$

a serwer B

$$R_B = 1 + (M + 1) * \sum_{i=1}^{M \cdot K - 1} \frac{(M \cdot K - 1)!}{(M \cdot K - i - 1)!} \quad \text{dla } M * K > 1, \quad (3)$$

przy czym całą różnicę, tj. $M * \sum_{i=1}^{M \cdot K - 1} \frac{(M \cdot K - 1)!}{(M \cdot K - i - 1)!}$, wysłał w segment nie zawierający serwerów (podsieć 300).

Ponieważ analizowane przykłady obejmują tylko wybrane fragmenty konfiguracji sieci, w celu umożliwienia dokładnej analizy liczby ramek generowanych w rzeczywistych wariantach konfiguracji sieci opracowano program symulatora *nbstorm.c*, który pozwala na symulację propagacji ramek rozgłoszeniowych w sieci o dowolnej, podanej topologii.

Dla zdefiniowania topologii należy każdej podsieci, jak i każdemu serwerowi przypisać unikalny numer, przy czym jest dopuszczalne istnienie serwera i podsieci o identycznym numerze. Warunki symulacji określa się przy pomocy pliku konfiguracyjnego, opisującego konfigurację symulowanej sieci.

W pliku konfiguracyjnym należy umieścić pary liczb X, Y oddzielone spacjami. Pierwsza liczba (X) identyfikuje serwer, a druga (Y) to numer podsieci. Taka linia oznacza, że serwer X jest podłączony do podsieci Y. Taki format pliku konfiguracyjnego pozwala na proste wprowadzanie bardzo skomplikowanych topologii sieci.

Użytkownik programu może wybrać jeden z dwóch sposobów rozpoczęcia eksperymentu.

Pierwszy zakłada, że jeden z serwerów sieci chce rozesłać ramkę rozgłoszeniową w protokole NetBIOS do wszystkich podsieci, do których jest on podłączony. W takim przypadku użytkownik powinien podać po opcji `"-server"` numer serwera rozpoczynającego transmisję. Drugi sposób polega na tym, że jedna ze stacji roboczych lub serwer nie objęty modelem (będący tylko źródłem ramek) przesyła ramkę rozgłoszeniową do jednej z podsieci, skąd jest ona następnie rozprowadzana przez serwery podłączone do tej podsieci. Użytkownik uruchamia ten tryb za pomocą opcji `"-net"` i numeru podsieci, w której rozpoczyna się symulacja.

Program po wczytaniu pliku konfiguracyjnego buduje w pamięci topologię sieci przez stworzenie list dwóch obiektów: serwerów i podsieci, przy czym dla każdego obiektu (np. serwera) jest dołączona lista obiektów innego typu (czyli np. podsieci), z którymi jest on połączony. Wszystkie obiekty są wyposażone w liczniki ramek odebranych, wysłanych, przesłanych lub zniszczonych. Struktura ta jest tworzona jednokrotnie na początku pracy programu, a symulacja powielania się liczby ramek rozgłoszeniowych odbywa się poprzez rekurencyjne wywoływanie procedur przetwarzających ramki wysłane i odbierane przez serwery.

Przykładowa topologia (zawartość pliku `ex1.txt`):

```
1 101
1 102
2 101
2 102
2 103
```

Uruchomienie symulacji rozpoczętej wysłaniem ramek z serwera 2:

```
nbstorm -server 2 ex1.txt
```

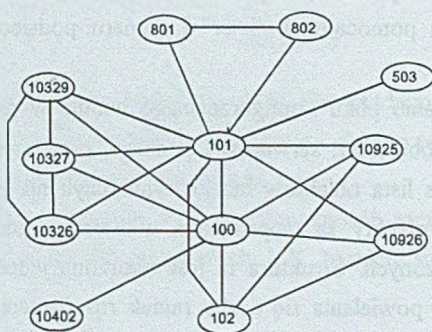
Uruchomienie symulacji rozpoczętej wysłaniem ramki do sieci 101:

```
nbstorm -net 101 ex1.txt
```

Wynikiem działania programu są informacje o liczbie przesłanych ramek przez każdy segment sieci oraz o liczbie wysłanych ramek przez każdego serwera.

5. Badania topologii i eksperymenty pomiarowe

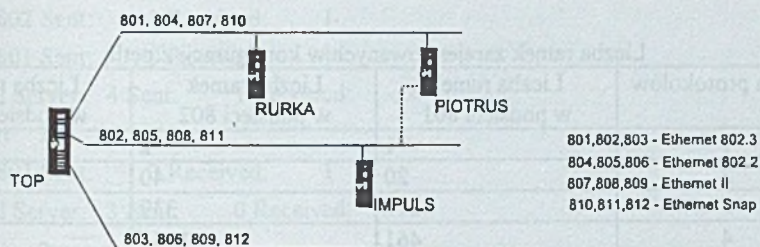
W celu określenia dokładnej topologii sieci próbowano posłużyć się programem wykorzystującym protokół diagnostyczny Novella, o nazwie *mwcare*, ale okazało się, że posiada on poważne ograniczenie, mianowicie ma ograniczoną w raportach do czterech liczbę numerów podsieci związanych z pojedynczym serwerem, nie można było więc przy kilku kartach sieciowych w serwerze odczytać wszystkich „zabindowanych” numerów podsieci. Ostatecznie informacje te odczytano, za zgodą osób administrujących, z plików konfiguracyjnych serwerów. Otrzymany obraz połączeń występujących pomiędzy numerami podsieci sieci wydziałowej przedstawia rys. 7.



Rys. 7. Połączenia pomiędzy podsieciami sieci wydziałowej

Fig. 7. Links among department LAN subnets

Ponieważ zawierał on wiele nałożonych na siebie pętli, postanowiono zamodelować w laboratorium fragment sieci i zbadać doświadczalnie liczbę generowanych w modelowej sieci ramek oraz powstające przy tym obciążenie serwerów. W eksperymentach wykorzystano serwery *TOP* i *IMPULS* Zakładu oraz dodatkowe serwery laboratoryjne, utworzone dla potrzeb tego zadania: *PIOTRUS* i *RURKA*. Schemat modelowanej sieci przedstawia rys. 8.



Rys. 8. Konfiguracja testowanego modelu sieci

Fig. 8. Network model prepared for test

W trakcie eksperymentów generowano programowo w segmencie 801 ramki zgłoszenia nazwy protokołu *NetBIOS* i mierzono przy pomocy analizatorów protokołów liczbę ramek pojawiających się w poszczególnych segmentach sieci przy zainstalowanych: *jednym, dwóch, trzech i czterech* protokołach liniowych dla każdej z kart sieciowych serwerów oraz przy działających dwóch (*TOP* i *IMPULS*), trzech (*TOP*, *IMPULS* i *RURKA*) i czterech serwerach. Zmierzone liczby ramek przeliczono na 1 ramkę „pobudzającą”.

Tabela 1

Liczba ramek zarejestrowanych w konfiguracji liniowej

Liczba serwerów	Liczba protokołów	Liczba ramek w podsieci 801	Liczba ramek w podsieci 802	Liczba ramek w podsieci 803
2	2	4	4	6
	3	15	15	21
	4	112	64	148
3	2	5	8	12
	3	45	86	141
	4	348	561	559
4	2	6	12	18
	3	79	211	345
	4	367	488	531

W trakcie pomiarów przy zainstalowanych trzech i więcej protokołach obserwowano „gubienie” ramek przez analizatory (pojawiały się komunikaty o błędach). W pierwszej fazie pomiarów badano konfigurację „liniową”, następnie wprowadzono zaznaczone linią kropkowaną połączenie zamykające pętlę i powtórzono testy. Wyniki pomiarów zebrano w tabelach 1 i 2.

Tabela 2

Liczba ramek zarejestrowanych w konfiguracji z pętlą

Liczba protokołów	Liczba ramek w podsieci 801	Liczba ramek w podsieci 802	Liczba ramek w podsieci 803
1	1	2	2
2	20	40	74
3	217	339	666
4	461	1251	886

Przedstawione w tabelach wartości „współczynnika powielania” ramek przy 3 i 4 protokołach nie są w pełni dokładne, ze względu na gubienie części ramek przez analizatory programowe (w trakcie eksperymentów niemal równocześnie generowane były przez serwery duże liczby ramek, statystyki serwerów wykazały gwałtowny wzrost liczby kolizji oraz liczby błędów typu „Brak wolnych bloków ECB”). Dla oceny tego zjawiska i porównania wyników przeprowadzono przy pomocy programu *nbstorm* symulację ujętych w obu tablicach przypadków.

Program symulatora rozróżnia kierunki przepływu ramek i osobno wyliczał ramki odebrane i wysłane przez każdy serwer w każdą z podsieci, jak również sprawdzał przekroczenie przez ramkę dopuszczalnego czasu życia. Ponieważ analizator odbierał ramki przesyłane torem fizycznym, niezależnie od kierunku ich transmisji, w celu umożliwienia porównywalności wyników należało zsumować odpowiednio ramki wyliczone dla poszczególnych podsieci danego łącza fizycznego (kable) oraz ograniczyć się przy sumowaniu tylko do ramek np. wysyłanych przez poszczególne serwery.

Poniżej przedstawiono oryginalny wydruk z symulatora dla pojedynczego eksperymentu, odpowiadającego konfiguracji pętli z 4 serwerami i 1 protokołem liniowym.

1 801

1 802

1 803

2 802

3 801

4 801

4 802

Nets:

Net 803 Transferred: 2

Net 802 Transferred: 2

Net 801 Transferred: 1

Servers:

Server 4

Net 802 Sent:	1	Received:	1
Net 801 Sent:	0	Received:	1
Total Server 4 Sent:	1	Received:	2
Server 3			
Net 801 Sent:	0	Received:	1
Total Server 3 Sent:	0	Received:	1
Server 2			
Net 802 Sent:	0	Received:	2
Total Server 2 Sent:	0	Received:	2
Server 1			
Net 803 Sent:	2	Received:	0
Net 802 Sent:	1	Received:	1
Net 801 Sent:	0	Received:	1
Total Server 1 Sent:	3	Received:	2
Total Sent:	4		
Total Received:	7		
Total Transferred:	5		
Total TooDeep:	0		

Poniżej przedstawiono przeliczone te same wyniki do postaci porównywalnej z wynikami eksperymentów:

1 801 802 803	(serwer TOP)
2 802	(serwer IMPULS)
3 801	(serwer RURKA)
4 801 802	(serwer PIOTRUS)

Nets:

Net 803 Transferred:	2
Net 802 Transferred:	2
Net 801 Transferred:	1

Poniżej w tablicach 3 i 4 zestawiono wyniki symulacji powielania 1 ramki rozgłoszeniowej dla obu badanych konfiguracji z wynikami pomiarów (w nawiasach).

Tabela 3

Wyniki symulacji liczby ramek dla konfiguracji liniowej

Liczba serwerów	Liczba protokołów	Liczba ramek w podsieci 801	Liczba ramek w podsieci 802	Liczba ramek w podsieci 803
2	2	4 (4)	4 (4)	x (6)
	3	15 (15)	15 (15)	21 (21)
	4	112 (112)	64 (64)	148 (148)
3	2	5 (5)	8 (8)	12 (12)
	3	45 (45)	87 (86)	141 (141)
	4	1363 (348)	1600 (561)	3520 (559)
4	2	6 (6)	12 (12)	18 (18)
	3	79 (79)	219 (211)	345 (345)
	4	3970 (367)	7360 (488)	11692 (531)

Tabela 4

Wyniki symulacji liczby ramek dla konfiguracji z pętlą

Liczba protokołów	Liczba ramek w podsieci 801	Liczba ramek w podsieci 802	Liczba ramek w podsieci 803
1	1 (1)	2 (2)	2 (2)
2	20 (20)	40 (40)	74 (74)
3	1207 (217)	1902 (339)	5664 (666)
4	145702 (461)	194576 (1251)	379708 (886)

Zwraca uwagę duża rozbieżność wyników symulacji i pomiarów dla czterech zainstalowanych protokołów w obu wersjach konfiguracji oraz dla trzech przy konfiguracji z pętlą. Ponieważ dla mniejszej liczby protokołów wyniki pomiarów i symulacji pokrywają się, można wyciągnąć stąd wniosek, że w przypadku idealnie szybko działających serwerów i analizatorów protokołów również i w tych przypadkach otrzymalibyśmy zbliżone liczby ramek.

W rzeczywistym przypadku lawiny ramek generowanych przez badany algorytm nie są w stanie wytrzymać ani serwery (stąd duża liczba błędów braku bloków ECB – ze względu na brak zasobów nie są generowane wszystkie wynikające z algorytmu ramki, i duża liczba kolizji), ani analizatory protokołów, działające na słabym sprzęcie (Pentium 75 w nitce 802, pozostałe dwa – Pentium 166) – stąd błędy gubienia ramek.

6. Wnioski i informacje końcowe

Z powyższych rozważań i przeprowadzonych pomiarów można wysnuć wniosek, że głównym odpowiedzialnym za pogorszenie się własności eksploatacyjnych sieci lokalnych na Wydziale są błędy konfigurowania oprogramowania routerów, polegające na niepotrzebnej

instalacji dodatkowych protokołów liniowych dla kart sieciowych w serwerach. Występowanie fizycznych pętli połączeń (kabli) sieciowych dodatkowo pogarsza sytuację, choć samo (przy jednym protokole) nie powoduje zauważalnego wzrostu obciążenia sieci. Błędy te ujawniają się bardzo wyraźnie przy protokołach stosujących często transmisje rozgłoszeniowe (*NetBIOS, RIP, SAP*).

Wymienione protokoły są protokołami technicznymi, wbudowanymi w działanie systemów operacyjnych – *RIP, SAP* w Netware do wersji 3.12, w wersji 4.x opcjonalne, *NetBIOS* we wszystkich wersjach Windows od 3.1 do NT; między innymi protokołu tego używają algorytmy udostępniania zasobów systemu (dyski, drukarki), rozgłaszając swoją usługę co kilkadziesiąt sekund w sieci.

Omawiane błędy konfiguracji nie dotyczą wyłącznie systemu Netware, można je również z identycznym skutkiem popełnić przy konfigurowaniu serwera NT, w którym także można dla jednej karty sieciowej zainstalować do czterech protokołów liniowych: *802.2, 802.3, Ethernet II oraz Ethernet SNAP*. W przeprowadzonych eksperymentach serwer NT retransmitował wszystkie odbierane na podsieci 802 ramki protokołów *RIP* i *SAP* kolejno w podsieci 805, 808 i 811, a udział transmisji typu broadcast w całkowitym ruchu pakietów w sieci sięgał 25 do 30%.

Dla przeprowadzenia przedstawianych w niniejszym artykule eksperymentów wykorzystano okres wakacyjny i pomagali przy ich realizacji w ramach praktyki studenci IV roku. Wyniki przeprowadzanych pomiarów oraz wynikające z nich wnioski rozeszły się „pocztą pantoflową” po zainteresowanych tym osobach, i do początków października br. osoby administrujące serwerami na podsieciach 103xx, 109xx i innych zmieniły konfigurację serwerów i badane zjawiska (*burze broadcastowe*) zniknęły. Obecnie udział ramek rozgłoszeniowych w całkowitym ruchu pakietów w sieci nie przekracza 1%.

LITERATURA

1. IPX Upgrade for NetWare Servers 1.0, Installation and configuration guide. Novell Inc., 100-002431-001, October 13, 1994.
2. The Importance of Network Design. <http://developer.novell.com/research/appnotes/1991-august/01/02.htm>.
3. The IPX/SPX Router Specification. Novell, Inc., Orem, Utah 1988.
4. Wolisz A.: Podstawy lokalnych sieci komputerowych. WNT, Warszawa 1992, tom 2, Oprogramowanie komunikacyjne i usługi sieciowe.
5. Tanenbaum A. S.: Sieci komputerowe. WNT, Warszawa 1988.

6. Kułaga P., Kopiec K., Rus L.: Badanie zjawiska nadmiernego powielania ramek broadcastowych protokołu NetBIOS w sieci lokalnej Wydziału Automatyki Elektroniki i Informatyki, Sprawozdanie z praktyki dyplomowej, sierpień 1998.

Recenzent: Dr inż. Ryszard Winiarczyk

Wpłynęło do Redakcji 21 kwietnia 1999 r.

Abstract

The article discusses issue existing in complex topology LAN. Broadcast frames duplicate and cause broadcast storm. Huge quantities of frames are transmitted in the net, stopping its normal way of working. These situations depend on network topology (fig. 5, 6). The article discusses real topology (fig. 1), used protocols (fig. 2, 3, 4) and mathematical model of this problem. (eq. 1, 2, 3). Results of and experiments are presented in tables. The optimal LAN topology is presented.