Krótki opis działania funkcjonalności Vidmon do monitoringu ruchu multicast:

Transmisja multicastowa w obecnych czasach nabiera coraz większego znaczenia, szczególnie w przypadku firm, które zajmują się tworzeniem / dostarczaniem treści audio/video. Transmisja takich danych stawia duże wymogi w stosunku do jakości sieci transmisyjnych. Dodatkowym problemem jest tutaj pomiar jakości sygnału. Wystarczą niewielkie fluktuacje opóźnienia, bądź zgubienie dosłownie kilku pakietów i jest to już widoczne u odbiorców końcowych. Poważnym problemem z punktu widzenia osób odpowiedzialnych za działanie sieci jest stwierdzenie faktu czy transmisja jest realizowana poprawnie, czy też wymaga poprawy. Pomimo, że idea transmisji multicast nie jest nowa, okazuje się, że są problemy z poprawną obsługą tego ruchu poprzez oferowane na rynku urządzenia. Problemy występują w sprzęcie zarówno małych, niszowych producentów, jak i w sprzęcie sieciowych gigantów. Mogą one przybierać czasami bardzo dziwne formy, np. konwerter, który nie przesyła ruchu multicast. Ruch wychodzi z routera, dociera do konwertera i ginie, albo ciche odrzucanie losowo wybranych ramek na karcie wyjściowej z routera. Podane przykłady to tylko część z problemów z którymi miałem okazje się spotkać, ale dobrze obrazują fakt, że transmisja multicast, to trudne zagadnienie. Bardzo trudne jest też diagnozowanie problemów takiej transmisji. Jest ona realizowana najczęściej w oparciu o protokół UDP i nie ma żadnych mechanizmów, które pozwalały by sterować ruchem (np. na zasadzie okna przesuwnego jak w TCP, lub informacji zwrotnej o zatorze albo ilości odrzuconych ramek). Są co prawda rozwiązania transmisji połączeniowej multicast, lecz do celów przesyłania strumieni audio/video się one nie nadają ze względu na charakter tej transmisji. Dodatkowo wolumen ruchu multicastowego często jest bardzo duży – przykładowo transmisja około 120 kanałów telewizyjnych zakodowanych kodekiem MPEG-2 enkapsulowanych w pakietach IP zajmuje około 600 – 700 Mb/s. Aby analizować ruch o tak dużym wolumenie wymagane są specjalistyczne mierniki. Oczywiście najlepszym sposobem monitorowania są użytkownicy końcowi, czyli np. w przypadku transmisji audio/video osoby oglądające daną transmisję (program), ale w przypadku transmisji ponad 100 kanałów nie można sobie pozwolić na zatrudnienie 100 osób do monitoringu, nie jest to opłacalne finansowo i daje jedynie informacje na temat, czy cała transmisja przebiega poprawnie, czy nie – a w przypadku gdy nie przebiega poprawnie – nie mamy możliwości sprawdzenia gdzie jest źródło problemu, czy jest to uszkodzony router, czy też sygnał wychodzi już uszkodzony ze źródła. Innym rozwiązaniem są specjalizowane mierniki, które pozwalają śledzić na bieżąco jakość sygnału, ale znowu – jeśli realizujemy transmisję do kilkudziesięciu lokalizacji docelowych, dodatkowo przez skomplikowaną sieć transmisyjną – zakup takiego zestawu mierników, który pozwolił by prześledzić poprawność transmisji na całej trasie to ogromny koszt. I właśnie tutaj przychodzi nam z pomocą rozwiązanie Cisco VIDMON (Inline Video Monitoring). Jest to rozwiązanie wymagające użycia konkretnych modeli routerów oraz kart liniowych, pozwalających na taki monitoring. Ja miałem okazje testować to rozwiązanie na routerze Cisco 7606 wyposażonym w kartę ES+ no i oczywiście stosowną wersję IOS. Postaram się w kilku słowach przybliżyć opis działania rozwiązania Vidmon. Przede wszystkim należy pamiętać, że jest ono skierowane do użycia przez administratorów sieci i ustalenia czy podczas transmisji multicastu występują jakieś problemy, czy też nie, a jeśli występują to w której części sieci. Nie należy traktować tego oprogramowania jako czegoś co zastąpi specjalistyczne mierniki – przynajmniej jeszcze nie na obecnym etapie ;-).
Na początek trochę o konfiguracji, wygląda ona identycznie jak dla QoS-a:

```
ip access-list extended TS_ALL
 permit udp host 192.168.105.66 any
 permit udp host 192.168.105.70 any
 permit udp host 192.168.105.74 any
 permit udp host 192.168.105.82 any

class-map match-any TS_ALL
 match access-group name TS_ALL
```

```
policy-map type performance-traffic vidmon_ts_all
 class TS_ALL
  monitor parameters
   description Warunki testowe #1
   history 180
   timeout 3
  monitor metric mdi
   rate media 41234166 bps
  monitor metric ip-cbr
   rate layer3 packet 3609.433 pps

interface GigabitEthernet1/3
 ip address 172.17.146.26 255.255.255.252
 ip pim sparse-mode
service-policy type performance-traffic output vidmon_ts_all
end
```

Lista dostępu TS_ALL wskazuje, co chcemy mierzyć. W moim przypadku wskazałem w niej urządzenia, które generują ruch multicastowy, czyli źródła. Dalej „class-map", która będzie używana do wskazanie klasy ruchu na który wskazuje lista dostępu i „policy-map" nowego typu „performance-traffic". Najwięcej problemu sprawiło mi podczas przygotowywania konfiguracji ustalenie parametrów transmisji. Muszą one być zdefiniowane bardzo dokładnie, jeśli chcemy otrzymać poprawne wyniki. I tak – w „policy-map" w sekcji „monitor parameters" definiujemy:

df – algorytm, wg którego będą przeliczane wyniki. Mamy do wyboru dwa: ipdv oraz rfc4445,
history – ilość próbek, które będzie pamiętał router wstecz,
interval – czas trwania pojedynczej próbki (domyślnie 30s),
timeout – ilość próbek, które trzeba odczekać po zaniku transmisji nim router przestanie monitorować dany transport,

W sekcji „monitor metric mdi" definiujemy co najmniej parametr „rate media", który określa dokładną przepływność danego transportu multicastowego. Uwaga – transportu a nie grupy. Grupa to transmisja adresowana na konkretny adres multicastowy – np. 224.10.10.101. Natomiast w ramach grupy multicastowej mogą być przesyłane różne transporty np. na różnych portach docelowych, czyli 224.10.10.101:7001, 224.10.10.101:7002, 224.10.10.101:7003 to różne transporty w ramach jednej grupy multicastowej. Podajemy przepływność pojedynczego transportu. W moim przypadku przepływność wszystkich transportów jest taka sama – około 38Mb/s, jeśli była by różna, musiał bym zdefiniować więcej klas w „policy-map".
W sekcji „monitor metric ip-cbr" podajemy co najmniej parametr „rate layer3 packet", jest to odpowiednik przepływności, ale wyrażony w pakietach na sekundę. Dotyczy – jak sama nazwa wskazuje pakietów w warstwie 3. W moim przypadku 1 pakiet IP niesie 7 pakietów MPEG po 188 B każdy + nagłówek UDP (8 B) + nagłówek IP (20 B), co daje wielkość 1 pakietu: 1344 B, przynajmniej tak mówi teoria oraz tak pokazują snify ;-), w praktyce okazało się, że aby uzyskać poprawne wyniki, a przynajmniej zbliżone do poprawnych – trzeba było nieco te parametry zmodyfikować. Jeśli podzielimy przepływność przez ilość pakietów, to wyjdzie nam, że wielkość pojedynczego pakietu powinna wynosić 1428 B. Jestem na etapie wyjaśniania tych rozbieżności z kolegami z działu DTV ;-).

Ostatnie, co trzeba zrobić, to przypisać „policy-map" do interfejsu, lub interfejsów, które chcemy monitorować.

Co możemy zobaczyć i zmierzyć (jeśli wszystko działa poprawnie:

```
vidmon#sh policy-map type performance-traffic int g1/3
 GigabitEthernet1/3

  Service-policy output: vidmon_ts_all

   class-map: TS_ALL
--------------------------------------------------------------------------------
Mon-Interval(sec): 30,  History(intvls): 180,  Timeout(sec): 90,  DF: rfc4445,  Total Flows: 22
--------------------------------------------------------------------------------

Flow: 0001, IPV4;   Dest: 224.10.10.119  Port: 7001;   Src: 192.168.105.74  Port: 4000
Agg Value(Per Flow): MDC: 96401,  MLR: 210460,  Pkt_cnt: 1058703481, MRV(%): 0.00000

Intvl           Updated at                Type    Pkt_cnt      MRV(%)/MLR   DF(msec)      MDC
-----+-----------------------------------+----+-------------+------------+----------+--------------
9781    22:02:59.537 MET Sun Apr 24 2011   cbr        108283      0.00000      0.409           NA
9781    22:02:59.537 MET Sun Apr 24 2011   mdi        108283            0      0.409            0
9780    22:02:29.537 MET Sun Apr 24 2011   cbr        108283      0.00000      0.413           NA
9780    22:02:29.537 MET Sun Apr 24 2011   mdi        108283            0      0.413            0
9779    22:01:59.537 MET Sun Apr 24 2011   cbr        108283      0.00000      0.408           NA
9779    22:01:59.537 MET Sun Apr 24 2011   mdi        108283            0      0.408            0
9778    22:01:29.537 MET Sun Apr 24 2011   cbr        108283      0.00000      0.410           NA
9778    22:01:29.537 MET Sun Apr 24 2011   mdi        108283            0      0.410            0
9777    22:00:59.541 MET Sun Apr 24 2011   cbr        108283      0.00000      0.414           NA
9777    22:00:59.541 MET Sun Apr 24 2011   mdi        108283            0      0.414            0
9776    22:00:29.541 MET Sun Apr 24 2011   cbr        108282      0.00000      0.416           NA
9776    22:00:29.541 MET Sun Apr 24 2011   mdi        108282            0      0.416            0
9775    21:59:59.541 MET Sun Apr 24 2011   cbr        108283      0.00000      0.423           NA
9775    21:59:59.541 MET Sun Apr 24 2011   mdi        108283            0      0.423            0
9774    21:59:29.540 MET Sun Apr 24 2011   cbr        108284      0.00000      0.430           NA
9774    21:59:29.540 MET Sun Apr 24 2011   mdi        108284            0      0.430            0
9773    21:58:59.536 MET Sun Apr 24 2011   cbr        108282      0.00000      0.423           NA
9773    21:58:59.536 MET Sun Apr 24 2011   mdi        108282            0      0.423            0
9772    21:58:29.536 MET Sun Apr 24 2011   cbr        108283      0.00000      0.413           NA
9772    21:58:29.536 MET Sun Apr 24 2011   mdi        108283            0      0.413            0
9771    21:57:59.536 MET Sun Apr 24 2011   cbr        108283      0.00000      0.417           NA
9771    21:57:59.536 MET Sun Apr 24 2011   mdi        108283            0      0.417            0
9770    21:57:29.536 MET Sun Apr 24 2011   cbr        108283      0.00000      0.414           NA
9770    21:57:29.536 MET Sun Apr 24 2011   mdi        108283            0      0.414            0

Flow: 0002, IPV4;   Dest: 224.10.10.118  Port: 7001;   Src: 192.168.105.74  Port: 4000
Agg Value(Per Flow): MDC: 23116,  MLR: 38871,  Pkt_cnt: 1058782632, MRV(%): 0.00000

Intvl           Updated at                Type    Pkt_cnt      MRV(%)/MLR   DF(msec)      MDC
-----+-----------------------------------+----+-------------+------------+----------+--------------
9781    22:03:00.697 MET Sun Apr 24 2011   cbr        108283      0.00000      0.428           NA
9781    22:03:00.697 MET Sun Apr 24 2011   mdi        108283            0      0.428            0
9780    22:02:30.697 MET Sun Apr 24 2011   cbr        108283      0.00000      0.436           NA
9780    22:02:30.697 MET Sun Apr 24 2011   mdi        108283            0      0.436            0
9779    22:02:00.697 MET Sun Apr 24 2011   cbr        108282      0.00000      0.435           NA
9779    22:02:00.697 MET Sun Apr 24 2011   mdi        108282            0      0.435            0
9778    22:01:30.697 MET Sun Apr 24 2011   cbr        108284      0.00000      0.444           NA
9778    22:01:30.697 MET Sun Apr 24 2011   mdi        108284            0      0.444            0
9777    22:01:00.701 MET Sun Apr 24 2011   cbr        108282      0.00000      0.442           NA
9777    22:01:00.701 MET Sun Apr 24 2011   mdi        108282            0      0.442            0
9776    22:00:30.701 MET Sun Apr 24 2011   cbr        108283      0.00000      0.437           NA
9776    22:00:30.701 MET Sun Apr 24 2011   mdi        108283            0      0.437            0
9775    22:00:00.701 MET Sun Apr 24 2011   cbr        108283      0.00000      0.437           NA
9775    22:00:00.701 MET Sun Apr 24 2011   mdi        108283            0      0.437            0
9774    21:59:30.700 MET Sun Apr 24 2011   cbr        108283      0.00000      0.428           NA
9774    21:59:30.700 MET Sun Apr 24 2011   mdi        108283            0      0.428            0
9773    21:59:00.696 MET Sun Apr 24 2011   cbr        108283      0.00000      0.441           NA
9773    21:59:00.696 MET Sun Apr 24 2011   mdi        108283            0      0.441            0
9772    21:58:30.696 MET Sun Apr 24 2011   cbr        108283      0.00000      0.436           NA
9772    21:58:30.696 MET Sun Apr 24 2011   mdi        108283            0      0.436            0
9771    21:58:00.696 MET Sun Apr 24 2011   cbr        108283      0.00000      0.434           NA
9771    21:58:00.696 MET Sun Apr 24 2011   mdi        108283            0      0.434            0
9770    21:57:30.696 MET Sun Apr 24 2011   cbr        108283      0.00000      0.421           NA
9770    21:57:30.696 MET Sun Apr 24 2011   mdi        108283            0      0.421            0
```

lub jeśli działa niepoprawnie:

```
vidmon#sh policy-map type performance-traffic int g1/3
 GigabitEthernet1/3

  Service-policy output: vidmon_ts_all

   class-map: TS_ALL
--------------------------------------------------------------------------------
Mon-Interval(sec): 30,  History(intvls): 180,  Timeout(sec): 90,  DF: rfc4445,  Total Flows: 22
--------------------------------------------------------------------------------

Flow: 0001, IPV4;   Dest: 224.10.10.119  Port: 7001;   Src: 192.168.105.74  Port: 4000
Agg Value(Per Flow): MDC: 119616,  MLR: 255046,  Pkt_cnt: 1061281910, MRV(%): 0.00000

Intvl           Updated at                Type    Pkt_cnt      MRV(%)/MLR   DF(msec)      MDC
-----+-----------------------------------+----+-------------+------------+----------+--------------
9805    22:14:59.539 MET Sun Apr 24 2011   cbr        108033     -0.23000     13.974           NA
9805    22:14:59.539 MET Sun Apr 24 2011   mdi        108033          488     13.974          265
9804    22:14:29.539 MET Sun Apr 24 2011   cbr        108024     -0.23900     13.976           NA
9804    22:14:29.539 MET Sun Apr 24 2011   mdi        108024          511     13.976          293
9803    22:13:59.539 MET Sun Apr 24 2011   cbr        107963     -0.29500     17.546           NA
9803    22:13:59.539 MET Sun Apr 24 2011   mdi        107963          715     17.546          374
9802    22:13:29.543 MET Sun Apr 24 2011   cbr        107689     -0.54800     38.332           NA
9802    22:13:29.543 MET Sun Apr 24 2011   mdi        107689         1585     38.332          766
9801    22:12:59.543 MET Sun Apr 24 2011   cbr        106682     -1.47800    104.807           NA
```

```
9801    22:12:59.543 MET Sun Apr 24 2011   mdi        106682         4721    104.807         2185
9800    22:12:29.542 MET Sun Apr 24 2011   cbr        106147     -1.97200    178.208           NA
9800    22:12:29.542 MET Sun Apr 24 2011   mdi        106147         5060    178.208         2629
9799    22:11:59.542 MET Sun Apr 24 2011   cbr        101942     -5.85500    427.238           NA
9799    22:11:59.542 MET Sun Apr 24 2011   mdi        101942        14921    427.238         7740
9798    22:11:29.542 MET Sun Apr 24 2011   cbr        102279     -5.54400    415.893           NA
9798    22:11:29.542 MET Sun Apr 24 2011   mdi        102279        10981    415.893         6049
9797    22:10:59.542 MET Sun Apr 24 2011   cbr        107326     -0.88300    117.275           NA
9797    22:10:59.542 MET Sun Apr 24 2011   mdi        107326         1750    117.275          864
9796    22:10:29.646 MET Sun Apr 24 2011   cbr        108086     -0.18100     12.573           NA
9796    22:10:29.646 MET Sun Apr 24 2011   mdi        108086          393     12.573          214
9795    22:09:59.537 MET Sun Apr 24 2011   cbr        108151     -0.12100      8.687           NA
9795    22:09:59.537 MET Sun Apr 24 2011   mdi        108151          273      8.687          141
9794    22:09:29.541 MET Sun Apr 24 2011   cbr        108160     -0.11300      7.576           NA
9794    22:09:29.541 MET Sun Apr 24 2011   mdi        108160          252      7.576          114
9793    22:08:59.541 MET Sun Apr 24 2011   cbr        108127     -0.14400      8.426           NA
9793    22:08:59.541 MET Sun Apr 24 2011   mdi        108127          285      8.426          152
9792    22:08:29.541 MET Sun Apr 24 2011   cbr        108113     -0.15600      9.797           NA
9792    22:08:29.541 MET Sun Apr 24 2011   mdi        108113          307      9.797          174
9791    22:07:59.540 MET Sun Apr 24 2011   cbr        107947     -0.31000     21.983           NA
9791    22:07:59.540 MET Sun Apr 24 2011   mdi        107947          693     21.983          369
9790    22:07:29.540 MET Sun Apr 24 2011   cbr        107890     -0.36200     25.032           NA
9790    22:07:29.540 MET Sun Apr 24 2011   mdi        107890          783     25.032          442

Flow: 0002, IPV4;   Dest: 224.10.10.118  Port: 7001;   Src: 192.168.105.74  Port: 4000
Agg Value(Per Flow): MDC: 27227,  MLR: 46438,  Pkt_cnt: 1061379846, MRV(%): 0.00000

Intvl           Updated at                Type    Pkt_cnt      MRV(%)/MLR   DF(msec)      MDC
-----+-----------------------------------+----+-------------+------------+----------+--------------
9805    22:15:00.699 MET Sun Apr 24 2011   cbr        108205     -0.07200      5.115           NA
9805    22:15:00.699 MET Sun Apr 24 2011   mdi        108205          385      5.115          216
9804    22:14:30.699 MET Sun Apr 24 2011   cbr        108188     -0.08700      5.398           NA
9804    22:14:30.699 MET Sun Apr 24 2011   mdi        108188          469      5.398          242
9803    22:14:00.699 MET Sun Apr 24 2011   cbr        108182     -0.09300      5.962           NA
9803    22:14:00.699 MET Sun Apr 24 2011   mdi        108182          488      5.962          267
9802    22:13:30.703 MET Sun Apr 24 2011   cbr        108164     -0.10900      7.892           NA
9802    22:13:30.703 MET Sun Apr 24 2011   mdi        108164          550      7.892          291
9801    22:13:00.703 MET Sun Apr 24 2011   cbr        108161     -0.11200      7.868           NA
9801    22:13:00.703 MET Sun Apr 24 2011   mdi        108161          558      7.868          301
9800    22:12:30.726 MET Sun Apr 24 2011   cbr        108184     -0.09100      6.501           NA
9800    22:12:30.726 MET Sun Apr 24 2011   mdi        108184          461      6.501          246
9799    22:12:00.702 MET Sun Apr 24 2011   cbr        108126     -0.14400      9.561           NA
9799    22:12:00.702 MET Sun Apr 24 2011   mdi        108126          726      9.561          407
9798    22:11:30.702 MET Sun Apr 24 2011   cbr        108139     -0.13200      8.444           NA
9798    22:11:30.702 MET Sun Apr 24 2011   mdi        108139          733      8.444          383
9797    22:11:00.702 MET Sun Apr 24 2011   cbr        108142     -0.13000      7.590           NA
9797    22:11:00.702 MET Sun Apr 24 2011   mdi        108142          657      7.590          363
9796    22:10:30.698 MET Sun Apr 24 2011   cbr        108165     -0.10800      8.699           NA
9796    22:10:30.698 MET Sun Apr 24 2011   mdi        108165          538      8.699          310
9795    22:10:00.697 MET Sun Apr 24 2011   cbr        108223     -0.05500      5.374           NA
9795    22:10:00.697 MET Sun Apr 24 2011   mdi        108223          319      5.374          157
9794    22:09:30.701 MET Sun Apr 24 2011   cbr        108245     -0.03500      3.435           NA
9794    22:09:30.701 MET Sun Apr 24 2011   mdi        108245          225      3.435          110
9793    22:09:00.701 MET Sun Apr 24 2011   cbr        108241     -0.03800      3.458           NA
9793    22:09:00.701 MET Sun Apr 24 2011   mdi        108241          221      3.458          122
9792    22:08:30.701 MET Sun Apr 24 2011   cbr        108242     -0.03700      3.751           NA
9792    22:08:30.701 MET Sun Apr 24 2011   mdi        108242          227      3.751          114
9791    22:08:00.700 MET Sun Apr 24 2011   cbr        108214     -0.06300      4.834           NA
9791    22:08:00.700 MET Sun Apr 24 2011   mdi        108214          326      4.834          184
9790    22:07:30.700 MET Sun Apr 24 2011   cbr        108197     -0.07900      5.945           NA
9790    22:07:30.700 MET Sun Apr 24 2011   mdi        108197          399      5.945          231
```

intvl – numer kolejnej próbki
updated at – dokładny czas próbki
type – rodzaj pomiaru (tutaj jedna próbka wyświetlana jest w dwóch liniach – jako mdi oraz cbr)
pkt_cnt – ile pakietów zostało złapanych z danego TS-u w czasie 1 próbki
MRV(%)/MLR – dla próbek cbr – to wartość odchyłki w % od zadeklarowanego pasma.
Wartość -0,363 oznacza, że używamy mniej pasma o wskazaną wartość od tego co zadeklarowaliśmy. Dla próbek mdi mamy podaną ilość utraconych pakietów, ale nie IP tylko MPEG.
DF (Delay Factor) - maksymalna różnica pomiędzy przychodzącymi pakietami (maksymalna wartość to 1000 ms)
MDC – ilość utraconych pakietów MPEG, ale mierzona w nieco inny sposób niż MLR, stąd obie wartości nie będą się pokrywać.

Co wart byłby jednak monitoring bez możliwości raportowania o problemach? Tylko zabawką, tak więc mamy narzędzia, które pozwolą wysłać nam komunikaty odpowiednio do sysloga, lub po SNMP. Konfigurujemy je jak niżej:

```
react 1 mdi-df
 alarm severity error
 threshold gt 0.600
```

react 2 mdi-mdc
alarm severity error
threshold gt 0

Generują one takie komunikaty, które informują nas o problemach:

```
Apr 24 22:37:33 172.18.18.6 2201: Apr 24 22:38:58.069: %FLOWMON-2-ALERT_ERROR_SET: [MDI-MDC]: SRC_IP:192.168.105.74,
SRC_PORT:4000, DS
T_IP:224.10.10.121, DST_PORT:7001, Gi1/3, Output (144)
Apr 24 22:37:33 172.18.18.6 2202: Apr 24 22:38:58.797: %FLOWMON-2-ALERT_ERROR_SET: [MDI-MDC]: SRC_IP:192.168.105.74,
SRC_PORT:4000, DS
T_IP:224.10.10.122, DST_PORT:7001, Gi1/3, Output (657)
Apr 24 22:37:34 172.18.18.6 2203: Apr 24 22:38:59.549: %FLOWMON-2-ALERT_ERROR_SET: [MDI-MDC]: SRC_IP:192.168.105.74,
SRC_PORT:4000, DS
T_IP:224.10.10.119, DST_PORT:7001, Gi1/3, Output (113)
Apr 24 22:37:36 172.18.18.6 2204: Apr 24 22:39:00.705: %FLOWMON-2-ALERT_ERROR_SET: [MDI-MDC]: SRC_IP:192.168.105.74,
SRC_PORT:4000, DS
T_IP:224.10.10.118, DST_PORT:7001, Gi1/3, Output (126)

Apr 24 22:38:41 172.18.18.6 2224: Apr 24 22:40:06.601: %FLOWMON-2-ALERT_ERROR_CLEAR: [MDI-MDC]: SRC_IP:192.168.105.74,
SRC_PORT:4000,
DST_IP:224.10.10.103, DST_PORT:7001, Gi1/3, Output (0)
Apr 24 22:38:41 172.18.18.6 2225: Apr 24 22:40:06.821: %FLOWMON-2-ALERT_ERROR_CLEAR: [MDI-MDC]: SRC_IP:192.168.105.70,
SRC_PORT:4000,
DST_IP:224.10.10.104, DST_PORT:7004, Gi1/3, Output (0)
Apr 24 22:38:41 172.18.18.6 2226: Apr 24 22:40:06.821: %FLOWMON-2-ALERT_ERROR_CLEAR: [MDI-MDC]: SRC_IP:192.168.105.70,
SRC_PORT:4000,
DST_IP:224.10.10.104, DST_PORT:7001, Gi1/3, Output (0)
Apr 24 22:38:41 172.18.18.6 2227: Apr 24 22:40:06.821: %FLOWMON-2-ALERT_ERROR_CLEAR: [MDI-MDC]: SRC_IP:192.168.105.70,
SRC_PORT:4000,
DST_IP:224.10.10.104, DST_PORT:7002, Gi1/3, Output (0)
```

Dzięki takiemu rozwiązaniu mamy narzędzie, które pozwala nam stwierdzić, czy problem z multicastem powstają w naszej Sieci podczas transmisji, czy też pojawiają się już u źródła. Łatwiej jest wtedy znaleźć przyczynę problemu i szybko ją usunąć.
Moim zamiarem w tym artykule było pokazanie nowego narzędzia dla administratorów sieci, jakim jest Vidmon. Nie zdążyłem go dokładnie opisać z braku czasu. Jeśli ktoś jest zainteresowany jego pełnymi możliwościami – odsyłam do Internetu na strony Cisco – np. tutaj:
http://www.cisco.com/en/US/docs/routers/7600/install_config/ES40_config_guide/es40_chap13.html

Napiszę jeszcze tylko, że porównywałem wskazania Vidmona z dedykowanymi miernikami multicastu i jego wskazania są zbliżone do wskazań mierników. Należy jednak mieć świadomość, że mierniki multicastu wykonują pomiary na znacznie mniejszych próbkach, przez co są znacznie dokładniejsze, ale w końcu – jak już zaznaczyłem – nie należy Vidmona uważać za coś co ma je zastąpić ;-).

A dla osób, które chciały by poczytać o tym co to takiego multicast – załączam krótki opis. Krótki, bo temat multicastu to temat na kilkutomową publikacje.

„Krótki" opis działania transmisji multicast

Wstęp:

Transmisja strumieni audio/video w sieciach komputerowych wymaga pewnych szczególnych rozwiązań. Jednym z takich szczególnych rozwiązań jest transmisja multicast. Oczywiście transmisja tego typu nie służy jedynie do przesyłania strumieni audio/video ale tak właśnie najczęściej ją się kojarzy. Ponadto przy transmisji strumieni audio/video pojawiają się pewne specyficzne problemy związane z wydajnością sieci oraz urządzeń. Na początek odpowiedź na pytanie – co takiego daje nam transmisja multicast, że warto ją zastosować? Przede wszystkim oszczędzamy pasmo w naszej sieci. Transmisja tego typu została zaprojektowana po to, aby dostarczać te same dane, w tym samym czasie z jednego źródła do wielu odbiorców. Skoro przesyłamy takie same dane, to mogą one być kopiowane w węzłach naszej sieci, gdzie transmisja rozchodzi się na kilka kierunków. Zacznijmy od małej powtórki:

Rodzaje ruchu:

- Unicast (IPv4, IPv6),
- Multicast (IPv4, IPv6),
- Broadcast (tylko IPv4),
- Anycast (tylko IPv6),

Ruch typu **unicast**, to ruch 1 do 1. Jedno urządzenie nadaje ramkę, która przeznaczona jest do jednego urządzenia docelowego. W nagłówkach warstw 2 i 3 stosowane są wtedy adresy umożliwiające jednoznaczne określenie urządzenia docelowego. Jeśli chcielibyśmy przesłać tę samą treść do kilku użytkowników końcowych przy użyciu ruchu typu unicast, musieli byśmy wysłać te same dane osobno do każdego z nich, co jest oczywiście wykonalne, ale nieefektywne.

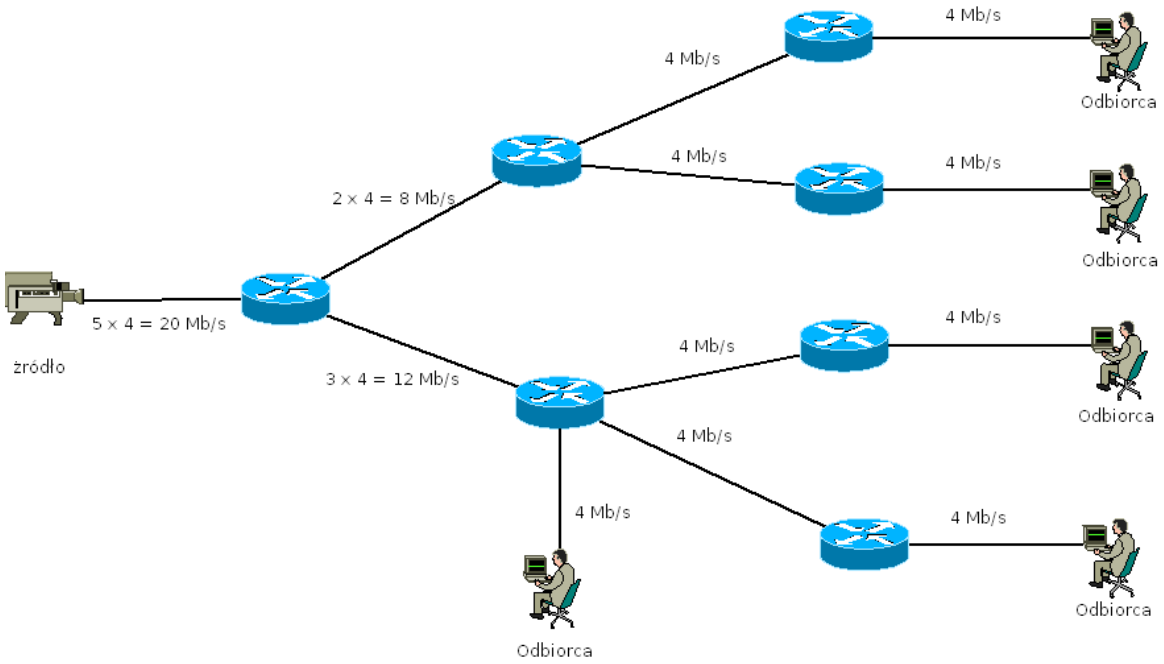
Ruch typu **multicast**, to ruch 1 do n, gdzie n jest dowolną liczbą węzłów, które wyraziły chęć otrzymywania ruchu multicast, poprzez wysłanie odpowiedniego żądania (IGMP w przypadku IPv4, MLD w IPv6). Ponieważ ruch kierowany jest do pewnej liczby odbiorców, o których źródło, czyli urządzenie nadające multicast nic nie wie, trzeba użyć specjalnie w tym celu przygotowanej adresacji. Została w tym celu przydzielona adresacja zarówno dla warstwy 2, jak i warstwy 3. Zostały też zdefiniowane sposoby mapowania adresacji pomiędzy warstwami. Ciekawostką jest to, że adresów L3 zarówno dla IPv4 jak i IPv6 jest znacznie więcej niż przypisanych adresów L2, więc mapowanie nie jest jednoznaczne.

Ruch typu **broadcast**, jest stosowany tylko w IPv4 i jest to ruch 1 do wszystkich. W odróżnieniu od ruchu typu unicast oraz multicast jest ograniczony tylko do sieci lokalnej (pojedynczej domeny broadcastowej) i nie jest rutowany. Ruch tego typu muszą odebrać wszystkie urządzenia w całej domenie, czyli w sieci lokalnej, dopiero router stanowi granicę dla ruchu tego typu. Powoduje to, że w dużych sieciach (z dużą ilością węzłów) ruch tego typu jest sporym problemem i nie pozwala na budowanie bardzo rozległych sieci w warstwie 2. W IPv6 zrezygnowano z transmisji broadcast zastępując go transmisją multicast.

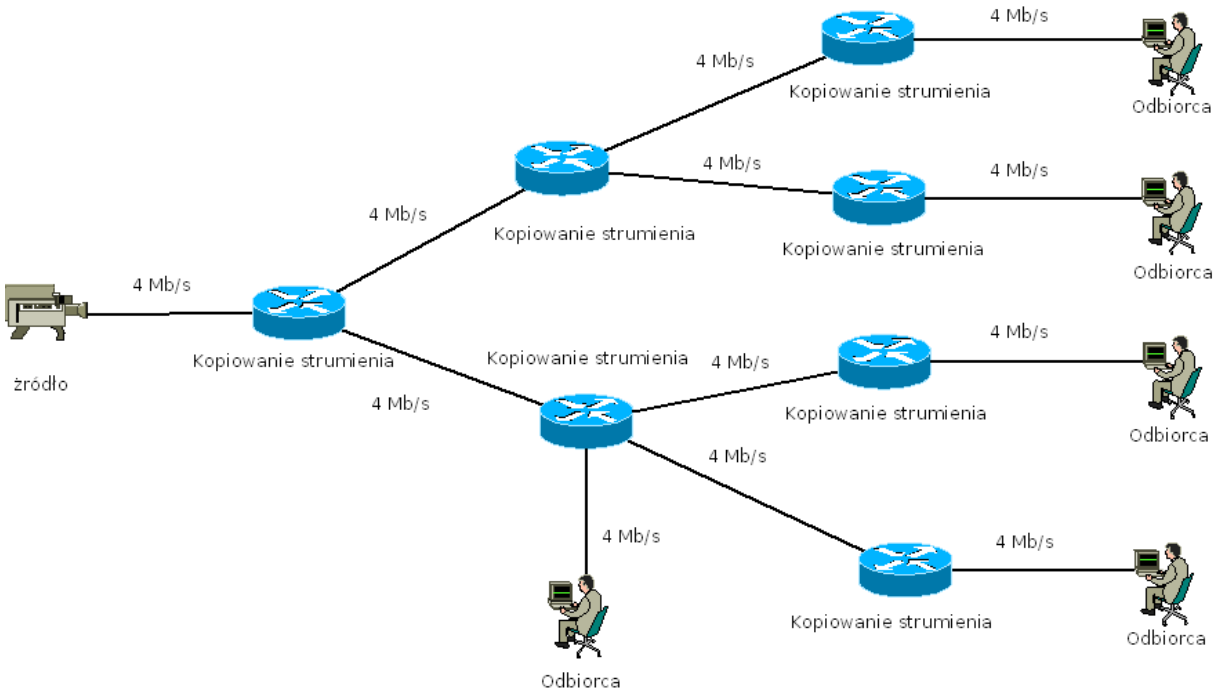
Ruch typu **anycast**, jest stosowany tylko w IPv6 i został wprowadzony po to, aby zwiększyć możliwości redundancji. Jeden i ten sam adres unicastowy jest przypisywany do kilku różnych urządzeń. Oczywiście urządzenia te powinny zostać poinformowane, że adres został przeznaczony do ruchu typu anycast, w przeciwnym przypadku zostanie zgłoszony konflikt adresów IP. Natomiast dla reszty urządzeń w sieci adresy anycast są normalnymi adresami, w efekcie urządzenia, które będą chciały przesłać dane na adres IPv6 anycast prześlą je do najbliższego urządzenia, które ma nadany taki adres. Można dzięki temu np. rozkładać ruch do/z Internetu na wiele routerów, albo równoważyć obciążenie serwerów.

Wróćmy teraz do wyjaśnienia dlaczego transmisja typu multicast nadaje się lepiej do transmisji strumieni audio/video niż np. unicast. Poniżej przykład transmisji np. pojedynczego kanału TV w jakości SD kodowanej MPEG-2 – zazwyczaj około 4 Mb/s. Załóżmy że mamy jedno źródło sygnału i przesyłamy ten sygnał do pięciu odbiorców w różnych częściach sieci. Transmisja unicast, to pięć strumieni – po jednym dla każdego odbiorcy. W takim przypadku źródło musi wysłać te same dane pięć razy, za każdym razem w polu adresu docelowego podając innego odbiorcę. Powoduje to, że

wzrost liczby odbiorców ma przełożenie na wzrost ruchu w sieci, szczególnie w bezpośrednim pobliżu źródła sygnału.



Może to doprowadzić do tego, że po przekroczeniu pewnej ilości odbiorców, kolejni nie będą w stanie połączyć się ze źródłem sygnału, bo brak będzie pasma (scenariusz optymistyczny), bądź podłączanie się kolejnych odbiorców zdegraduje jakość sygnału dla wszystkich (scenariusz pesymistyczny). Właśnie tutaj z pomocą przychodzi nam transmisja multicast. Działa ona w ten sposób, że dane ze źródła wysyłane są tylko jednym strumieniem, natomiast routery i inne urządzenia sieciowe, przez które ta transmisja przepływa są w stanie rozpoznać na których interfejsach wyjściowych znajdują się odbiorcy i kopiują strumień danych na te interfejsy. W efekcie końcowym ilość odbiorców nie ma wpływu na obciążenie sieci. Niezależnie czy dany strumień odbiera 5 czy 50 odbiorców – używana jest zawsze taka sama przepływność.



Zatem ten typ transmisji pozwala nam oszczędzać pasmo w sieci.

Adresacja:

Ponieważ źródło multicastu nie wie ilu odbiorców odbiera emitowany przez niego strumień danych, a tym bardziej nie ma pojęcia o adresach tych odbiorców, musiał zostać wprowadzony specjalny sposób adresowania. Zostały więc wprowadzone specjalnie do tego celu zarezerwowane grupy adresów w warstwie 3 oraz w warstwie 2.

IPv4 – zarezerwowana do celu multicastu została klasa D, czyli adresy IP zaczynające się od ciągu bitów 1110. Daje to zakres adresów od 224.0.0.0 do 239.255.255.255. W ramach tego zakresu adresów wprowadzono dodatkowe podziały.

Range	Mask	Description
224.0.0.0-224.0.0.255	224.0.0/24	Local Network Control Block
224.0.1.0-224.0.1.255	224.0.1/24	Internetwork Control Block
224.0.2.0-224.0.255.255	-	Ad hoc Block
224.1.0.0-224.1.255.255	-	Unassigned
224.2.0.0-224.2.255.255	224.2/16	SDP/SAP Block
224.3.0.0-231.255.255.255	-	Unassigned
232.0.0.0-232.255.255.255	232/8	Source Specific Multicast Block
233.0.0.0-233.255.255.255	233/8	GLOP Block
234.0.0.0-238.255.255.255	-	Unassigned
239.0.0.0-239.255.255.255	239/8	Administratively Scoped Block

Źródło: http://www.cisco.com/en/US/tech/tk828/technologies_white_paper09186a00802d4643.shtml

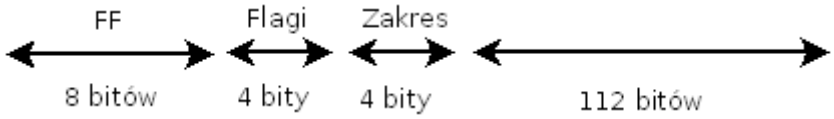
Oczywiście w ramach wskazanych zakresów występują jeszcze głębsze podziały. Zainteresowanych odsyłam do wskazanego źródła (http://www.cisco.com/en/US/tech/tk828/technologies_white_paper09186a00802d4643.shtml), lub bezpośrednio do RFC3171bis.

Następnie adresy multicast warstwy 3 mapowane są na adresy multicast warstwy 2. Do mapowania adresów multicastowych IPv4 na adresy MAC został przewidziany następujący blok tych adresów: 01:00:5E:00:00:00, zmienne są 23 najmłodsze bity. (Aż się prosi, żeby zapisać to jako: 01:00:5E:00:00:00/25 ;-). Mapowanie polega na przepisaniu 23 najmłodszych bitów adresu IP na 23 najmłodsze bity adresu MAC. Proszę jednak zwrócić uwagę na to, że w adresie warstwy 3 mamy zdefiniowane na stałe tylko 4 najstarsze bity. Wynika z tego, że pozostałe 28 bitów może przyjmować dowolne wartości, co daje 268 435 456 możliwych kombinacji. W adresie warstwy 2 mamy z kolei zdefiniowane na stałe 25 bitów, co oznacza że do mapowania pozostało nam 23 bity (czyli o 5 bitów mniej niż w adresie warstwy 3), co daje 8 388 608 możliwości. Oznacza to, że mapowania nie będą jednoznaczne. Oznacza także że do jednego adresu warstwy 2 będzie można zamapować 32 adresy warstwy 3. Przykład: następujące adresy IPv4 można zamapować do pojedynczego adresu MAC:

224.0.0.1, 224.128.0.1, 225.0.0.1, 225.128.0.1, 226.0.0.1, 226.128.0.1, 227.0.0.1, 227.128.0.1, 228.0.0.1, 228.128.0.1, 229.0.0.1, 229.128.0.1, 230.0.0.1, 230.128.0.1, 231.0.0.1, 231.128.0.1, 232.0.0.1, 232.128.0.1, 233.0.0.1, 233.128.0.1, 234.0.0.1, 234.128.0.1, 235.0.0.1, 235.128.0.1, 236.0.0.1, 236.128.0.1, 237.0.0.1, 237.128.0.1, 238.0.0.1, 238.128.0.1, 239.0.0.1, 239.128.0.1

Adres MAC, do którego w/w adresy będą mapowane: 01:00:5E:00:00:01. Należy o tym pamiętać przy przydzielaniu adresów multicastowych dla źródeł sygnału.

IPv6 – w nowej wersji protokołu IP na cele transmisji multicast zarezerwowano blok adresów: FF00::/8, przy czym ma on pewną specyficzną wewnętrzną strukturę:



Flagi składają się z 4 birów: 0RPT
R – RFC3956, Embedded RP
P – RFC3306, Multicast address butli based on a unicast prefix
T – RFC3513, Permanently assigned multicast address

Zakres – wskazuje routerom informacje potrzebne do obsługi multicastu wewnątrz właściwej domeny:
0001 Interface-local scope
0010 Link-local scope
0011 Subnet-local scope
0100 Admin-local scope
0101 Site-local scope
1000 Organization-local scope
1110 Global-local scope

Oczywiście, podobnie, jak w poprzednim przypadku adresy zostały wstępnie podzielone na kilka różnych zakresów. Zostało też zdefiniowanych kilka możliwości utworzenia adresu multicastowego. Ponieważ w IPv6 zrezygnowano z transmisji typu broadcast i zastąpiono ją transmisją multicast, zostały wydzielone specjalne adresy IP do tego celu, np:
Wszystkie hosty w zadanym zakresie: FF0X::1
Wszystkie routery w zadanym zakresie: FF0X::2
X – oznacza wybrany zakres wg wskazanego powyżej wzoru.
Adresy multicastowe powiązane z hostami (solicited-node): FF02::1:FFxx:xxxx, gdzie najmłodsze 24 bity są wstawiane w zależności od adresu unicast danego hosta.
Szczegółowe informacje na temat podziału adresów multicastowych IPv6 można znaleźć w RFC3513.
Oczywiście, tak samo jak w przypadku IPv4 musi istnieć możliwość mapowania adresów IPv6 na odpowiadające im adresy MAC. Realizowane jest to bardzo podobnie, jak w przypadku IPv4, inny jest natomiast przydzielony zakres adresów MAC. Jest to zakres od 33:33:00:00:00:00 do 33:33:FF:FF:FF:FF, gdzie mapowane są najmłodsze 32 bity adresu IPv6 na najmłodsze 32 bity adresu MAC. Tak samo jak w przypadku IPv4 adresów warstwy 2 jest mniej, niż adresów warstwy 3, tylko w tym przypadku różnica jest o wiele większa: 2^88. Przykładowe mapowanie adresu IPv6 na adres MAC:
IPv6: FF02::1:FF5C:F038 -> MAC: 33:33:FF:5C:F0:38

Przedstawiony powyżej opis adresacji tylko bardzo pobieżnie porusza ten temat, ale pozwala zorientować się w używanych zakresach adresów i rozpocząć własne testy multicastu.

Zasada działania:

Transmisja multicastu jest skomplikowanym zadaniem. Postaram się go przybliżyć w dwóch krokach. Po pierwsze jak wygląda komunikacja pomiędzy odbiorcą znajdującym się w sieci lokalnej a pierwszym routerem. Po drugie jak wygląda komunikacja pomiędzy routerami w sieci. Zanim zacznę zagłębiać się w szczegóły zaznaczę od tego, że routery Cisco nie obsługują routingu ruchu multicast domyślnie, jeśli chcemy skorzystać z tej możliwości, musimy ją świadomie włączyć. Ponadto ruch multicast nie jest obsługiwany w sieci Internet, co nie pozwala na wykorzystanie dobrodziejstw, które oferuje w skali światowej.

Krok pierwszy – komunikacja pomiędzy odbiorcą a bezpośrednio podłączonym routerem:

Komunikacja ta ma na celu poinformować router, że w sieci lokalnej znajduje się co najmniej jeden aktywny odbiorca, który jest zainteresowany odbiorem wskazanej grupy multicastowej. Grupa multicastowa to transmisja realizowana przy użyciu pojedynczego adresu multicast. Może to być pojedyncza transmisja, lub grupa transmisji (np. na różnych portach docelowych), ale ważne jest to, że transmisje są realizowane na takim samym adresie multicast IP (adres docelowy. Źródłowy adres, to adres urządzenia nadającego, czyli źródła multicastu). To, czy w sieci lokalnej znajduje się jeden, czy więcej odbiorców, nie ma znaczenia dla routera. Dla niego ważna jest tylko informacja czy ma transmitować daną grupę multicastową do wskazanego segmentu sieci, czy nie – a do tego celu wystarczy informacja, że jest tam co najmniej jeden aktywny odbiorca. W przypadku IPv4 do komunikacji pomiędzy Odbiorcami a routerem służy protokół IGMP (Internet Group Management Protocol), w IPv6 IGMP został zastąpiony przez MLD (Multicast Listener Discovery). IGMP to mechanizm, który pozwala hostom informować bezpośrednio podłączony router, które grupy multicastowe chce mieć przyłączone lub odłączone. Natomiast router multicastowy używa IGMP do sprawdzenia, czy jacyś odbiorcy multicastu są podłączeni w sieciach, w których ma on bezpośrednio interfejsy. Niezależnie od wersji protokół IGMP działa z grubsza tak samo:

- jeśli w segmencie sieci, w której router ma interfejs nie ma odbiorców multicastu, router nie wysyła na ten interfejs żadnej transmisji multicastowej.
- jeśli któryś z hostów w sieci lokalnej chce stać się aktywnym odbiorcą jakiejś grupy multicastowej, wysyła komunikat IGMP: host membership report do routera, w którym podaje, jaką grupę chciał by otrzymywać.
- po otrzymaniu IGMP: host membership report, router rozpoczyna transmisję wskazanej grupy multicastowej do wskazanej podsieci. Jednocześnie to na routerze spoczywa odpowiedzialność pilnowania, czy w tej podsieci cały czas znajdują się aktywni odbiorcy. Router realizuje ten cel wysyłając zapytania IGMP: host membership query. Jeśli w danej podsieci są aktywni odbiorcy multicastu, odpowiadają wysyłając komunikat IGMP: host membership report (ten sam, co przy podłączaniu się do grupy multicastowej). Obecnie protokół IGMP występuje w trzech wersjach:

IGMP v.1 (RFC1112)

IGMP jest enkapsulowane w datagramie IP i ma ustawiony protokół numer 2. Jeśli host chce zostać odbiorcą wskazanej grupy multicastowej wysyła komunikat IGMP host membership report (bez komunikatu IGMP host membership query od strony routera). Następnie podczas całej transmisji odpowiada na zapytania IGMP host memberhip query ze strony routera. Gdy natomiast host chce zrezygnować z odbioru grupy multicastowej, po prostu przestaje odpowiadać na zapytania ze strony routera. Router, jeśli nie otrzyma odpowiedzi na kilka kolejnych zapytań, przyjmuje, że w segmencie nie ma już aktywnych odbiorców i przestaje transmitować wskazaną grupę multicastową.

IGMP v.2 (RFC2236)

Jest kompatybilna z wersją 1 i zawiera ulepszenia w stosunku do niej, na przykład:

- obsługa routerów w sieciach z wielodostępem. Jeśli do jednego segmentu podłączonych jest kilka routerów, to tylko jeden z nich wysyła IGMP host membership query. Pozostałe routery słyszą odpowiedzi na zapytania. Router, który wysyła zapytania jest wybierany na podstawie najniższego adresu IP.
- wersja druga daje też odbiorcom możliwość aktywnego zrezygnowania z odbioru wskazanej grupy multicastowej. Służy do tego celu komunikat IGMP host membership leave.

IGMP v.3 (RFC3376)

Zostało stworzone do obsługi protokołu PIM Source Specific Mode (będzie omówiony dalej). Zasadniczy powód jest taki, że w komunikatach IGMP host membership report, oprócz grupy multicastowej, którą odbiorca chce otrzymywać przesyła routerowi informacje na temat adresu IP źródła, z którego chce tą grupę otrzymywać.

W IPv6 protokół IGMP został zastąpiony przez protokół MLD. Protokół MLD w odróżnieniu od IGMP jest wysyłany w ramkach ICMPv6, poza tym działa identycznie jak protokół IGMP. MLD v.1 = IGMP v.2, MLD v.2 = IGMP v.3. MLD dodatkowo posiada funkcjonalności, których IGMP nie miało – np.

rozszerzenie AAA (Authentication, Authorization, Accounting), które pozwala na autoryzacje i rozliczanie odbiorców multicastu.

Zanim przejdę do szybkiego omówienia protokołów routingu multicastu pomiędzy routerami, chciałem jeszcze zwrócić uwagę na problemy, które mogą się pojawić podczas transmisji multicastu w sieciach LAN bazujących na przełącznikach. Problem polega na tym, że przełączniki działają w sposób, który opiera się na wyznaczeniu portu wyjściowego w oparciu o docelowy adres MAC zawarty w ramce. Przełącznik uczy się na którym porcie znajduje się jakie urządzenie analizując adres źródłowy ramki. Ramka multicastowa w polu docelowego adresu MAC zawiera adres wyliczony na podstawie adresu IP. Jest to adres MAC wirtualny, dodatkowo nigdy na przełączniku nie pojawi się ramka, która miała by taki adres w polu źródłowym, w efekcie czego przełącznik nigdy nie nauczy się pary port – adres MAC multicastowy. W przypadku, gdy przełącznik nie wie, gdzie wysłać ramkę, bo nie ma przyporządkowania MAC-port w tablicy przełączania, to wysyła taką ramkę na wszystkie porty z wyjątkiem tego, na którym przyszła. Tak też traktowany jest cały ruch multicastowy, co niepotrzebnie zajmuje pasmo w całym segmencie. Istnieje kilka możliwości aby tę sytuacje naprawić:
- ręcznie wprowadzić adres MAC do tablicy przełączania i powiązać go z portem na którym znajduje się odbiorca. Metoda dobra, ale nie dająca się skalować, ponadto zmiana grupy multicastowej wymusza ręczną rekonfiguracje wszystkich przełączników w sieci.
- uruchomienie pomiędzy switchem a routerem protokołu CGMP (Cisco Group Management Protocol). Jest to protokół firmowy Cisco. Jego działanie polega na tym, że router komunikuje się ze switchem i wysyła mu informacje potrzebne do tego, żeby określić na którym porcie podłączony jest odbiorca multicastu i jaką grupę multicastową odbiera. Na tej podstawie switch może wyliczyć MAC i dodać go do tablicy przełączania w parze z właściwym portem. Protokół CGMP dostępny jest tylko w IPv4, do IPv6 nie został przeniesiony.
- IGMP Snooping. Jest to rozwiązanie, które polega na tym, że switch podsłuchuje transmisje IGMP pomiędzy hostem a routerem i na tej podstawie określa potrzebne mu informacje. Nie jest wymagana współpraca z routerem, i jest to rozwiązanie otwarte. W IPv6 stosujemy analogicznie działające MLD Snooping.

Krok drugi – protokoły routingu stosowane pomiędzy routerami:

Zadaniem protokołów routingu multicastu, stosowanych na połączeniach pomiędzy routerami jest zbudować optymalną ścieżkę do transmisji multicastu pomiędzy źródłem a odbiorcami oraz uniknięcie pętli routingu. Ścieżka budowana jest na zasadzie drzewa MDT (Multicast Distribution Tree). Zadanie to nie jest proste, szczególnie jeśli uświadomimy sobie co jest do zrobienia. Z jednej strony sieci mamy źródło, które transmituje sygnał multicastowy do wszystkich odbiorców, z drugiej strony sieci mamy odbiornik/odbiorniki, które oznajmiają nam jaką grupę/grupy multicastową chcą odbierać. Zadaniem routera, który usłyszy żądanie IGMP membership report jest:
- określenie IP źródła multicastu, lub wskazanie innego routera, który może to zrobić,
- poinformowaniu najbliższego routera na trasie do źródła o potrzebie odbioru ruchu ze wskazanej grupy multicastowej.

Wyzwaniem jest znalezienie IP źródła multicastu. Adresem docelowym w pakietach przenoszących multicast jest IP grupy multicastowej, natomiast IP źródłowym jest standardowy adres IP urządzenia generującego ten ruch, aby zestawić ścieżkę transmisyjną trzeba umieć określić na podstawie adresu grupy multicastowej IP źródła transmisji. Różne protokoły routingu multicastu realizują to wyzwanie w różny sposób. Omówię bardziej szczegółowo działanie protokołu PIM (Protocol Independent Multicast) w różnych trybach pracy, aby przedstawić możliwe rozwiązania tego zadania. Protocol Independent oznacza, że PIM jest niezależny od protokołu routingu dynamicznego użytego dla ruchu unicast i może współpracować z każdym z nich.

PIM-DM – PIM-Dense Mode tak zwany tryb gęsty, został zaprojektowany do użycia w sieciach LAN, gdzie pasmo nie jest krytycznym zasobem. Występuje w dwóch wersjach (v.1 i v.2) pakiety v.1 są inkapsulowane w IGMP, natomiast v.2 jest przenoszona bezpośrednio w IP, numer protokołu 103. Zasada działania w obu przypadkach jest taka sama. Gdy w sieci pojawi się źródło multicastu i zacznie

wysyłać ruch, transmisja multicastowa jest przesyłana przez wszystkie routery na wszystkie interfejsy, które mają skonfigurowany protokół PIM. Następnie te interfejsy, na których nie ma aktywnych odbiorców są odcinane. W efekcie powstaje optymalne drzewo transmisji multicastu. Aby być pewnym, że multicast dociera do wszystkich odbiorców, regularnie powtarza się taką samą procedurę jak przy podłączeniu źródła. W przypadku PIM-DM, IP źródła nie ma znaczenia, bo transmisja jest rozprowadzana wszędzie, dopiero gdy routery zorientują się, że nie mają aktywnych odbiorców odcinają transmisję na niepotrzebnych interfejsach. Wadą tego rozwiązania jest to, że sieć jest regularnie zalewana niepotrzebnym ruchem. Zaletą rozwiązania jest jego prostota.
PIM-SM – PIM-Sparse Mode tak zwany tryb rzadki, został zaprojektowany do sieci WAN, gdzie pasmo jest bardzo cenne. Działa na zasadzie modelu explicite join, to znaczy, że żadna transmisja multicastowa nie jest realizowana bez jawnego żądania ze strony odbiorcy. W sieci wyznaczany jest router, który pełni funkcje miejsca spotkań „Randezvous Point" (RP), zawiera on informacje o adresach grup multicastowych oraz adresach IP powiązanych z nimi źródeł. W sieci może występować więcej niż jeden punkt spotkań (RP). Transmisja realizowana jest w ten sposób, że transmisja multicastu ze źródła jest przenoszona najkrótszą drogą do punktu spotkań (RP) i tam się kończy. Gdy Odbiorca wyśle komunikat IGMP host membership report do lokalnego routera, router przesyła tę informacje do punktu spotkań (RP), co powoduje, że multicast jest przesyłany z (RP) do odbiorcy. Tak zestawiona ścieżka ST (Shared Tree) nie musi być optymalna, więc po jej zestawieniu następuję optymalizacja do SPT (Shortest Path Tree), która ma na celu wyznaczenie optymalnej trasy pomiędzy źródłem a odbiorcą. Trasa ta może omijać punkt spotkań (RP). Dodatkowo w przypadku PIM-SM, musi być uruchomiony jeszcze router BSR (bootstrap router), może to być ta sama maszyna co RP, ale nie musi. Router BSR jest odpowiedzialny za propagowanie w sieci informacji o tym, która grupa multicastowa jest obsługiwana przez który RP.
PIM-SSM – Source Specific Multicast, został zaprojektowany jako uproszczenie PIM-SM. Tryb ten nie wymaga wyznaczenia punktu spotkań, nie budowane jest też drzewo ST (Shared Tree), a określenie IP źródła multicastu przerzucone zostało na odbiorce. Odbiorca musi w komunikacie IGMP host membership report oprócz grupy multicastowej podać IP źródła, budowane jest wtedy od razu optymalne drzewo SPT (Shortest Path Tree) do transmisji multicastu. Oznacza to, że PIM-SSM wymaga użycia IGMP v.3 / MLD v.2. Niestety okazuje się, że niewiele urządzeń ma zaimplementowaną obsługę jednego z tych protokołów, w związku z tym, aby umożliwić korzystanie z PIM-SSM tym urządzeniom wprowadzono inne możliwości określenia IP źródła ruchu multicastowego:
- ręczna konfiguracja mapowań na routerze (administrator podaje ręcznie pary: grupa multicastowa – IP źródła),
- mapowanie realizowane poprzez DNS – router po otrzymaniu komunikatu IGMP host membership report odpytuje serwer DNS o odpowiednio spreparowany rekord odwrotny (PTR), a w odpowiedzi otrzymuje IP źródła multicastu.
PIM-Bidir – został zaprojektowany do obsługi specyficznego ruchu multicast, jaki generują połączenia konferencje. Do tej pory mieliśmy do czynienia z jednym źródłem i wieloma odbiorcami. Natomiast w tym przypadku, każdy z członków konferencji odbiera dane i nadaje dane w ramach jednej grupy multicastowej. Mamy więc wielu odbiorców i wiele źródeł. W rozwiązaniu tym stosuje się, tak jak w PIM-SM punkt spotkań (RP), ale cała transmisja realizowana jest zawsze przez ten punkt spotkań, czyli w ramach niekoniecznie optymalnego drzewa ST (Shared Tree). Nie ma możliwości budowy drzewa SPT (Shortest Path Tree), ze względu na istnienie wielu źródeł potencjalnej transmisji. W trybie pracy PIM-Bidir nie jest też wykonywane sprawdzanie ścieżki RPF-check, ponieważ transmisja w ramach drzewa ST może być dwukierunkowa.
RPF check – transmisja multicastu powinna być wolna od pętli. Mechanizm RPF check ma za zadanie wyeliminować powstanie pętli. Jego działanie jest nadzwyczaj proste i wykorzystuje unicastową tablice routingu routera. Mechanizm ten zakłada, że ruch multicastowy będzie dostarczany do routera najkrótszą drogą. Każdy pakiet multicastowy zawiera w polu SRC IP adres IP urządzenia, które go wygenerowało. Router więc wykonuje sprawdzenie, czy interfejs, na którym otrzymał pakiet, jest tym samym interfejsem, który został by użyty do osiągnięcia IP źródła multicastu. Jeśli tak – pakiet

jest przyjmowany i przetwarzany. Jeśli nie – pakiet jest po cichu odrzucany. Ważną kwestią jest też kontrola zasięgu transmisji strumienia multicast. Używa się do tego celu pola TTL w nagłówku IP. A w przypadku IPv6 mamy jeszcze możliwość wpływania na zasięg transmisji poprzez wprowadzenie odpowiedniej wartości w 4-bitowe pole „zakres" w pakiecie multicastowym przewidziane do tego celu. Źródło transmitujące daną grupę decyduje o tym, jaki zasięg powinna mieć transmisja.
Na chwilę obecną wygląda na to, że największą popularność wśród protokołów routingu multicastu zdobył PIM. Należy jednak także pamiętać, że istnieją inne protokoły pozwalające na transmisję multicastu:
DVMRP – Distance Vector Multicast Routing Protocol. DVMRP używa pakietów IGMP z ustawionym polem typ na wartość 0x13. Stosunkowo stary protokół oparty o koncepcję dystans-vektor.
MOSPF – protokół OSPF wyposażony o dodatkowe rozszerzenia pozwalające na ruch multicast.
MBGP – protokół BGP wyposażony o dodatkowe rozszerzenia pozwalające na ruch multicast.
Do protokołu IPv6 przeniesiono tylko jeden protokół routingu multicastu – PIM. Ale zrezygnowano z trybu pracy Dense Mode. Dostępne są więc tylko trzy tryby pracy tego protokołu: Sparse-Mode, Source Specific Mode oraz Bidirectional.

Źródła:
William R. Parkhurst „Cisco Multicast Routing and Switching"
Ciprian Popoviciu, Eric Levy-Abegnoli, Patrick Grossetete „Deploying IPv6 Networks"
Internet:
Dokumentacja umieszczona na http://www.cisco.com
Baza danych dokumentów RFC

Paweł Kucharczyk
pawel@logruss.pl