1.1 Ewolucja systemów zarządzania sieciami IP

Główny wysiłek programistów i twórców protokołu w momencie powstawania sieci ARPANET koncentrował się na uzyskaniu pożądanych efektów komunikacyjnych i tylko w tym obszarze pojawiały się zaczątki rozwiązań zarządzających. Jedynym narzędziem dostępnym w fazie początkowej był protokół ICMP *(Internet Control Message Protocol)*. Oferowane przez niego mechanizmy nie pozwalają jednak na prawie nic więcej, niż transferowanie prostych komunikatów sygnalizujących występowanie problemów ze skutecznym transferowaniem treści informacyjnych. Jednakże implementacja ICMP ma charakter powszechny tj. realizuje go każdy element sieci opartej na protokole IP. Wykorzystanie wymienionych opcji ICMP wraz z dostępnymi funkcjami niektórych pól nagłówka pakietów IP pozwala na konstruowanie prostych choć użytecznych narzędzi zarządzających np. program PING *(Packet Internet Groper),* który może być wykorzystywany do potwierdzenia aktywności wybranych elementów sieci, a także wyznaczania opóźnienia komunikatów w zamkniętej pętli oraz poziomu strat wysyłanych datagramów. PING pozwala na określanie granic obszarów sieci, w których występuje stan natłoku oraz lokalizacji uszkodzonych połączeń. Uniwersalny charakter procedury PING pozwolił odsunąć potrzebę poważnego zajęcia się problematyką zarządzania środowiskiem internetowym aż do połowy lat osiemdziesiątych, kiedy gwałtowny przyrost użytkowników uczynił dotychczasowe rozwiązania niemożliwymi do dalszego wykorzystywania.

Szczególnie istotnym elementem rozwoju systemu stał się w tym czasie znaczący przyrost ilości podsieci owocujący zwiększeniem domen sieciowych administrowanych przez niezależne ośrodki decyzyjne. Pierwszym z długiej serii protokołów zarządzania środowiskiem systemowym było opracowane w 1987 rozwiązanie określane jako *Simple Gateway Monitoring Protocol (SGMP).* SGMP otwierał możliwość monitorowania bramek integrujących lokalne struktury sieciowe w kompleksowy system sieci rozległej. Równocześnie pojawiły się także i inne rozwiązania:

- *High-Level Entity Management System (HEMS)* - generalizacja pierwszego z używanych w Internecie protokołów zarządzania określanego mianem *Host Monироring Protocol (HMP)*.
- *Simple Network Management Protocol (SNMP)* - rozbudowana wersja SGMP.
- *CMIP over TCP/IP (CMOT)* - adaptacja rozbudowanego protokołu zarządzania środowiskiem OSI *(Common Management Information Protocol - CMIP),* dzięki której systemy internetowe uzyskały dostęp do funkcji, struktur baz danych zarządzania i innych elementów zgodnych z podejściem prezentowanym przez ISO.

W roku 1988 IAB dokonało przeglądu wymienionych rozwiązań, rekomendując wykorzystanie SNMP jako opcji doraźnej oraz CMOT jako docelowej. Założono, że wszelkie instalacje bazujące na zestawie TCP/IP zostaną z biegiem czasu przystosowane do wymagań sformułowanych przez OSI. Ułatwienie postulowanego przejścia stanowi formalne wymaganie aby zarówno SNMP, jak i CMOT korzystały z identycznych baz danych zarządzanych obiektów. Takie same są w szczególności zmienne sterujące i monitorujące hosty, routery, mostki i inne obiekty. Oznacza to całkowitą unifikację struktury informacji zarządzania *(Stucture of Management Information - SMI)*, a więc formatów opisowych obiektów oraz baz danych MIB. W rezultacie godzina „0” oznaczać będzie, że zmienią się protokoły (i oprogramowanie) ale wszystkie pozostałe elementy pozostaną takie same.

Wkrótce jednak okazało się, że przedstawione podejście trudno uznać za uzasadnione. Filozofia wyznawana przez ISO warunkuje traktowanie obiektów zarządzania jako rozbudowanych jednostek z licznym zestawem atrybutów, wzajemnie skojarzonych procedur i innych właściwości wynikających z

podejścia obiektowego. Tymczasem naturalna prostota SNMP wymaga rezygnacji z wymyślnych koncepcji na rzecz traktowania obiektów jako prostych zmiennych o kilku zaledwie charakterystycznych wartościach. Uznając przedstawioną argumentację za słuszną IAB złagodziło w roku 1989 swoje wymagania, zezwalając w efekcie, by rozwój SNMP i CMOT postępował niezależnie i równolegle. Uwolnienie SNMP z formalnych związków z wymaganiami OSI spowodowało wkrótce jego szybki choć często nie w pełni kontrolowany rozwój. W rezultacie wszyscy liczący się wytwórcy komputerów, stacji roboczych, mostków, routerów, hubów itp. oferują wraz ze swymi produktami systemy zarządzania oparte na SNMP.

Prowadzone są intensywne prace badawcze zmierzające do wprowadzenia SNMP do środowiska OSI oraz systemów wykorzystujących protokoły inne niż TCP/IP. Oczekuje się, że liczba kierunków, w których prowadzone są działania rozwojowe będzie stale rosnąć. Typowym przykładem nowej jakości w zarządzaniu opartym na SNMP jest idea zdalnego monitoringu stanowiąca podstawę formalnej specyfikacji znanej jako RMON *(Remore Monitoring)*. Rozszerzenie standardowej bazy MIB oraz związane z tym ściśle wprowadzenie nowych funkcji oddziaływania na jej zawartość, umożliwia sprawny monitoring podsieci jako całości bez potrzeby komunikowania się z indywidualnymi elementami użytkowymi.

Dostępne są również i inne rozszerzenia MIB, z których część stanowi niestandardowe rozwiązania oferowane przez indywidualnych producentów sprzętu, inne natomiast odpowiadają stosowanym od dawna technologiom w rodzaju Token Ring albo FDDI. Możliwości dalszego rozwijania SNMP jedynie poprzez kolejne rozszerzenia specyfikacji MIB są ograniczone. Dlatego opracowywane są kolejne wersje protokołu, które bazując zasadniczo na podstawowym schemacie realizacyjnym (SNMPv1) dostarczają nowych możliwości za cenę wykorzystania nieco bardziej złożonych mechanizmów. W chwili obecnej nowe wersje znane są jako SNMPv2 i SNMPv3.

1.1.1 SNMPv1 - reguły funkcjonowania

1.1.1.1 Architektura systemu zarządzania sieciowego

Model zarządzania wykorzystywany w sieciach z protokołami TCP/IP wykorzystuje następujące podstawowe elementy składowe:

- Stację zarządzającą *(Management station)*;
- Agenta zarządzania *(Management agent)*;
- Bazę danych informacji zarządzania *(Management Information Base - MIB)*;
- Protokół zarządzania sieciowego *(Network management protocol)*.

Stacja zarządzająca pełni funkcję interfejsu pomiędzy operatorem i systemem zarządzania zasobami systemowymi. Powinna być wyposażona przynajmniej w następujące elementy:

- Zestaw aplikacji zarządzania umożliwiających analizowanie danych, likwidację zagrożeń i uszkodzeń itp.;
- Interfejs pozwalający na monitorowanie i sterowanie funkcjonowaniem sieci;
- Funkcje translacji poleceń i komend operatora na akcje nadzorowania oddalonych elementów systemu;
- Bazę danych umożliwiającą przechowywanie informacji pozyskanych z baz MIB wszystkich zarządzanych elementów nadzorowanej sieci.

Innym aktywnym składnikiem systemu zarządzania jest aplikacja agenta, która uruchomiona w kluczowych elementach sieciowych typu hosty, routery, mostki i inne, odpowiada na żądania kierowane do nich ze stacji zarządzającej. W niektórych przypadkach możliwe jest również powiadamianie realizowane wyłącznie z inicjatywy agenta.

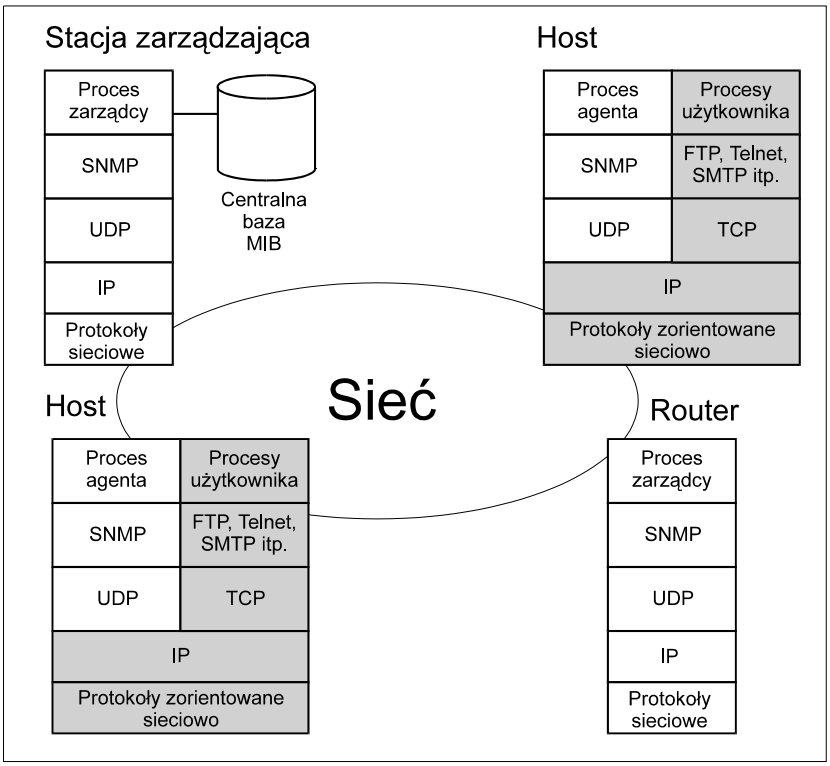
Całość zasobów systemu posiada reprezentację w postaci obiektów rozumianych jako zmienne reprezentujące istotne aspekty funkcjonalne. Zestaw tych zmiennych stanowi zawartość bazy MIB, która jest udostępniana stosownie do aktualnych potrzeb.

Obiekty są definiowane standardowo w ramach poszczególnych klas urządzeń (np. routery, mostki itp.), co umożliwia jednolite nadzorowanie elementów o różnych charakterystykach funkcjonalnych. Wszelkie akcje zarządzania wdraża się modyfikując wartości odpowiednich zmiennych obiektowych.

Stacja zarządzająca oraz aplikacja agenta porozumiewają się wykorzystując uzgodnioną procedurę nazywaną protokołem. Normatywy SNMP nie określają wymaganej ilości stacji zarządzających ani liczby aplikacji agenta przypadających na pojedynczą stację. Jednak dobrą praktyką jest posiadanie co najmniej dwóch stacji zarządzających, co umożliwia bezpieczne funkcjonowanie systemu w przypadku awarii. W kwestii liczby obsługiwanych aplikacji agenta obowiązuje zasada, że dopóki SNMP pozostanie rozwiązaniem relatywnie „prostym", ilość ta może być duża i wynosić setki aplikacji.

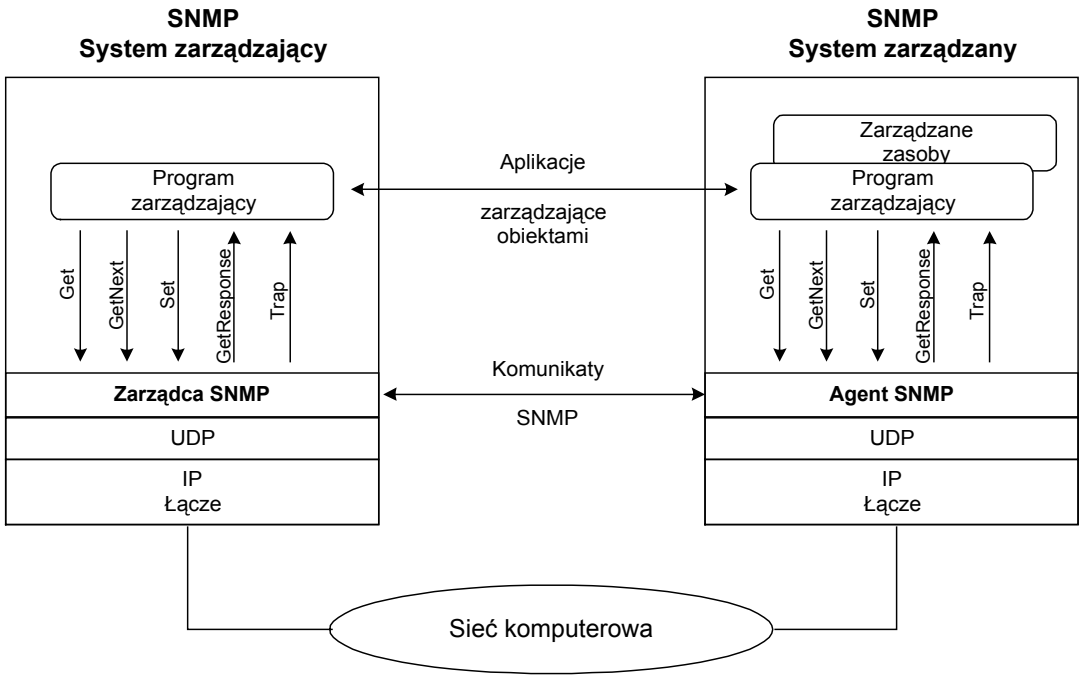
1.1.1.2 Współpraca elementów

SNMP stanowi element zestawu protokołów TCP/IP funkcjonujący w warstwie aplikacji i wykorzystujący mechanizmy dostępne w ramach UDP *(User Datagram Protocol)*. Procesy zarządzania wykorzystują również IP oraz lokalne protokoły systemu wynikające ze specyfiki konkretnej instalacji sieciowej (np. Ethernet, FDDI, X.25 i in.).



Konfiguracja funkcjonalna SNMP

Każdy z agentów wykorzystuje integralne mechanizmy SNMP, UDP i IP, interpretuje wiadomości zarządzania oraz nadzoruje własną bazę danych MIB. Jeśli agent funkcjonuje w urządzeniu realizującym usługi aplikacyjne w rodzaju FTP, powinien realizować nie tylko schemat UDP, ale także TCP.



Funkcje SNMP

Stacja zarządzająca może generować w imieniu aplikacji sterującej komunikaty, które są potwierdzane przez aplikację agenta. Agent może generować wiadomość typu `trap`, jeśli wystąpi wskazane zdarzenie systemowe.

Wykorzystanie przez SNMP datgramowego protokołu UDP oznacza, że wymiana informacji zarządzania odbywa się w trybie bezpołączeniowym. W rezultacie każdy przekaz stanowi indywidualną transakcję pomiędzy agentem i stacją zarządzającą.

1.1.1.3 Przeglądanie pułapkowane

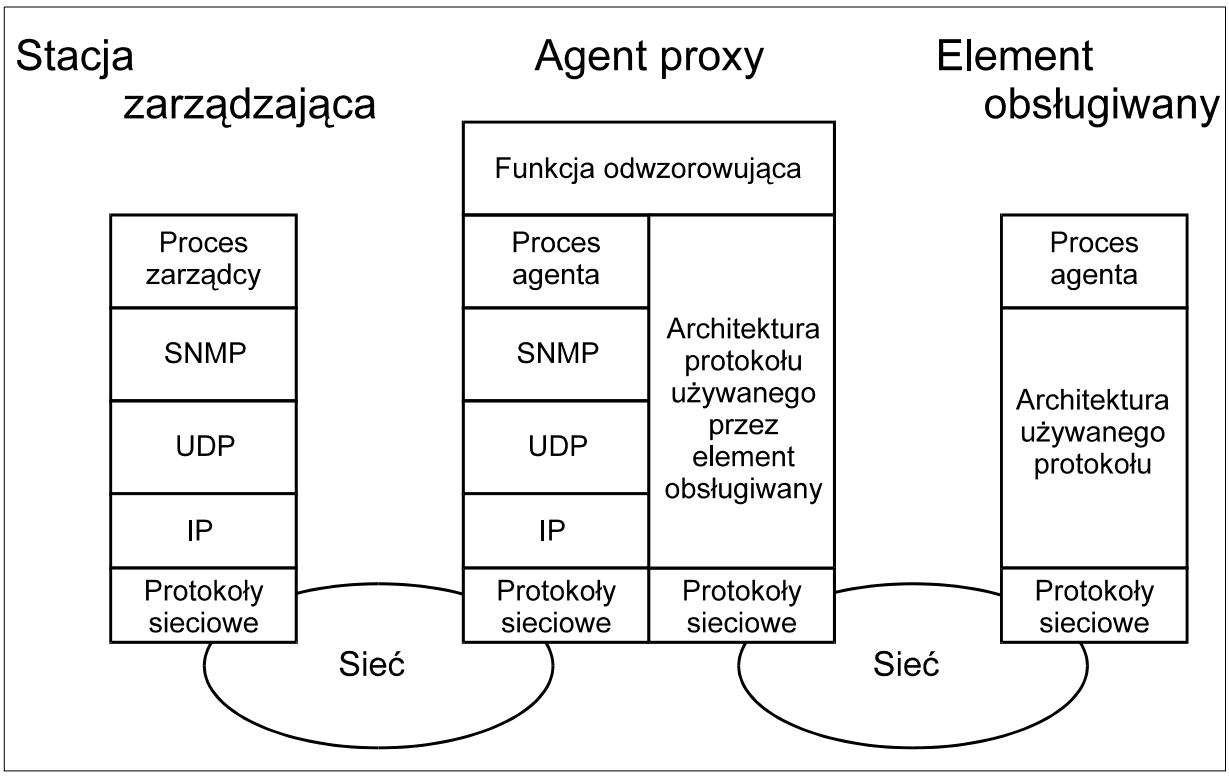
Jeśli stacja zarządzania nadzoruje dużą liczbę agentów, z których każdy obsługuje wiele obiektów, niemożliwa jest realizacja scenariusza regularnego odpytywania. W takim przypadku stosowana jest zmodyfikowana procedura, określana jako przeglądanie zorientowane na pułapkowanie *(trap-directed polling)*.

W trakcie inicjalizacji, a także co pewien ustalony odstęp czasu (np. raz dziennie), stacja zarządzająca odpytuje wszystkich agentów, żądając przekazania kluczowych informacji. Ustanowienie tak rozumianej ogólnej orientacji na temat zarządzanego systemu pozwala na rezygnację z periodycznego przeglądania, które zostaje zastąpione przez opcję indywidualnego informowania aplikacji zarządzającej przez poszczególnych agentów.

1.1.1.4 Aplikacje proxy

W systemach występują niekiedy urządzenia, które jakkolwiek korzystają z zestawu protokołów internetowych, to nie powinny być obciążane koniecznością realizacji mechanizmów SNMP, logiki agenta i utrzymaniem bazy danych MIB.

W celu włączenia urządzeń nie realizujących mechanizmów SNMP do zarządzanej domeny, wykorzystywana jest koncepcja proxy. W jej ramach pojedynczy agent SNMP reprezentuje jeden lub więcej elementów systemu, podejmując wszelkie niezbędne działania w ich imieniu.



Konfiguracja schematu proxy

Stacja zarządzająca przesyła zapytania dotyczące konkretnego urządzenia do jego agenta proxy, który transferuje ich treść na format obsługiwanego protokołu zarządzającego innego niż SNMP. Również otrzymane odpowiedzi są odpowiednio tłumaczone. Pośrednicząca aplikacja proxy odpowiada także za dostosowanie komunikatów o zdarzeniach *(event notifications)* do formy właściwej wiadomościom typu *trap*.

1.1.1.5 Środowisko rozproszone

Zarządzanie sieciowe jest realizowane w środowiskach rozproszonych, co oznacza, że ustanawia wzajemne interakcje wielu oddalonych jednostek systemowych obsługiwane przez wykorzystywany protokół.

W przypadku rozwiązań realizujących filozofię SNMP, jednostki aplikacyjne występują jako stacje zarządzające oraz aplikacje agenta. Z operacyjnego punktu widzenia, pozycja stacji zarządzającej jest uprzywilejowana w stosunku do podległych jej agentów.

Dopuscza się rozwiązania wykorzystujące większą liczbę stacji zarządzających, z których każda może nadzorować wszystkie albo tylko niektóre elementy podrzędne. Indywidualne domeny zarządzania mogą zawierać zatem jednostki wspólne (tj. podlegające więcej niż jednemu zarządcy równocześnie).

System SNMP można opisać jako zespół relacji pomiędzy pojedynczym agentem i zbiorem stacji zarządzających, którym jest podporządkowany. Szczególnie w tym przypadku agent powinien w sposób kompletny nadzorować własną bazę danych MIB. Nadzór ten realizowany jest w następujących aspektach:

- Uwierzytelniania *(authentication)*.
- Polityki dostępu *(access policy)*.
- Usług typu proxy *(proxy service)*.

Wszystkie wymienione aspekty nadzoru dotyczą bezpośrednio funkcji bezpieczeństwa.

SNMP w ujęciu RFC 1157 oferuje jedynie najprostrze i ograniczone możliwości wypełnienia funkcji bezpieczeństwa, wykorzystując podejście bazujące na pojęciu tzw. wspólnoty *(comrnunity)*.

Wspólnota SNMP stanowi zestaw składający się z agenta oraz zbioru stacji zarządzających, których wzajemne relacje obejmują: uwierzytelnianie, sterowanie dostępem oraz usługę proxy. Każda wymagana kombinacja wymienionych aspektów funkcjonalnych może być uzyskana poprzez powołanie oddzielnej wspólnoty, stanowiącej w efekcie koncepcję lokalną, definiowaną w systemie zarządzanym.

Agent może powołać wiele wspólnot, w tym z nakładającym się wzajemnie członkowstwem stacji zarządzających. Określenie wspólnoty następuje poprzez jej nazwę występującą w kontekście konkretnej jednostki zarządzania. Nazwa ta powinna być wykorzystywana podczas wdrażania operacji *get* i *set*, zaś z uwagi na lokalny charakter może być używana w obrębie systemu wielokrotnie (przez różne aplikacje agentów). W rezultacie identyczność nazw nie jest znacząca tj. nie wskazuje w szczególności na jakiekolwiek podobieństwa pomiedzy oznaczonymi przez nie wspólnotami. Stacja zarządzająca musi być zatem w stanie identyfikować nazwy wspólnot tworzonych przez różnych agentów, z którymi winna się komunikować.

1.1.1.6 Uwierzytelnianie

Jednostka SNMP odbierająca wiadomość powinna mieć możliwość upewnienia się, że jej źródłem jest rzeczywiście określony element systemu zarządzania. Zgodnie z definicją zawartą w RFC 1157, SNMP realizuje prosty schemat uwierzytelnienia, w którym dowolny komunikat przekazywany przez stację zarządzającą do agenta zawiera nazwę właściwej wspólnoty. Nazwa ta stanowi formę hasła, którego znajomość przez stronę inicjującą potwierdza jej uprawnienia.

Uwzględniając niski poziom bezpieczeństwa tak rozumianego uwierzytelnienia wielu twórców systemów zarządzania ogranicza jego zastosowanie do wdrażania funkcji monitoringu (operacji `GET` i `TRAP`). Rzeczywiste sterowanie nadzorowaną siecią, realizowane za pomocą funkcji `SET`, odwołuje się w takim przypadku do bardziej zaawansowanych realizacji korzystających np. z szyfrowania, które jednak w RFC 1157 nie zostały uwzględnione.

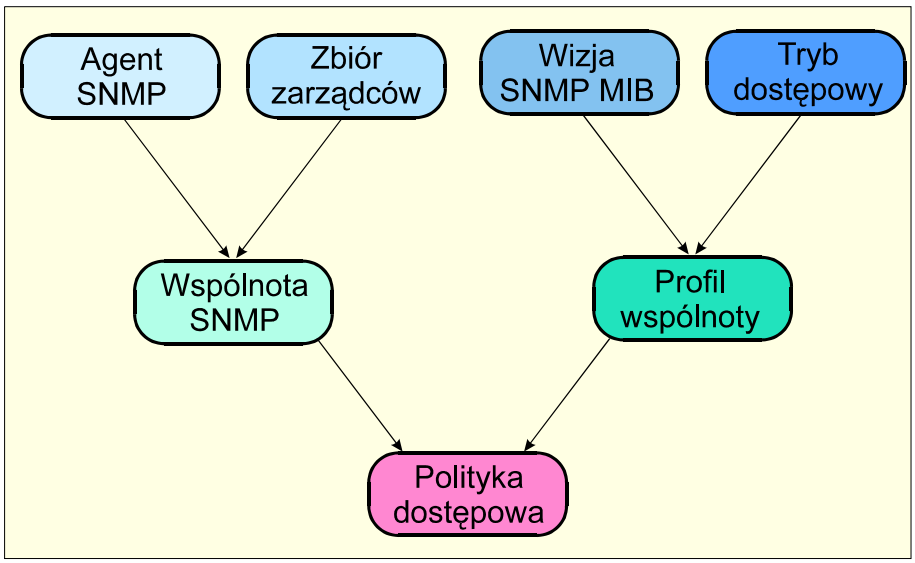
1.1.1.7 Polityka dostępowa

Polityka dostępu może być realizowana w następujących aspektach:

- Podzbioru udostępnianych obiektów MIB przypisywanego różnym wspólnotom. Zbiór obiektów w zestawie nie musi przy tym należeć do pojedynczego poddrzewa MIB.
- Trybu realizacji dostępu wyrażonego w kategoriach (`READ-ONLY, READ-WRITE`), które można definiować oddzielnie dla każdej wspólnoty.

Kombinacja wyboru obiektów oraz trybu ich udostępniania stanowi tzw. profil wspólnoty SNMP *(SNMP community profile).* Wskazany tryb dostępu dotyczy zawsze wszystkich wybranych obiektów i dlatego stacja zarządzająca przynależna do wspólnoty funkcjonującej w modzie `READ-ONLY` może jedynie realizować operacje odczytu.

Często używanym terminem jest tzw. polityka dostępowa SNMP *(SNMP access policy)* oznaczająca pewną wspólnotę wraz z określonym dla niej profilem.



Zależności pojęć podstawowych

1.1.1.8 Usługa proxy

Koncepcja wspólnoty bywa użyteczna podczas realizacji usługi proxy, czyli funkcjonowania agenta we wspólnym imieniu przypisanych mu, zewnętrznych elementów systemu. Elementy te stanowią najczęściej urządzenia, które nie realizują mechanizmów TCP/IP i SNMP, albo których interakcje z systemem zarządzania nie są pożądane.

Agent proxy realizuje politykę dostepową SNMP dla każdego reprezentowanego urządzenia, a więc musi być poinstruowany, które obiekty MIB powinien w tym celu wykorzystać (wybór) oraz w jaki sposób je udostępniać (tryb dostępu).

1.1.1.9 Specyfikacja protokołu

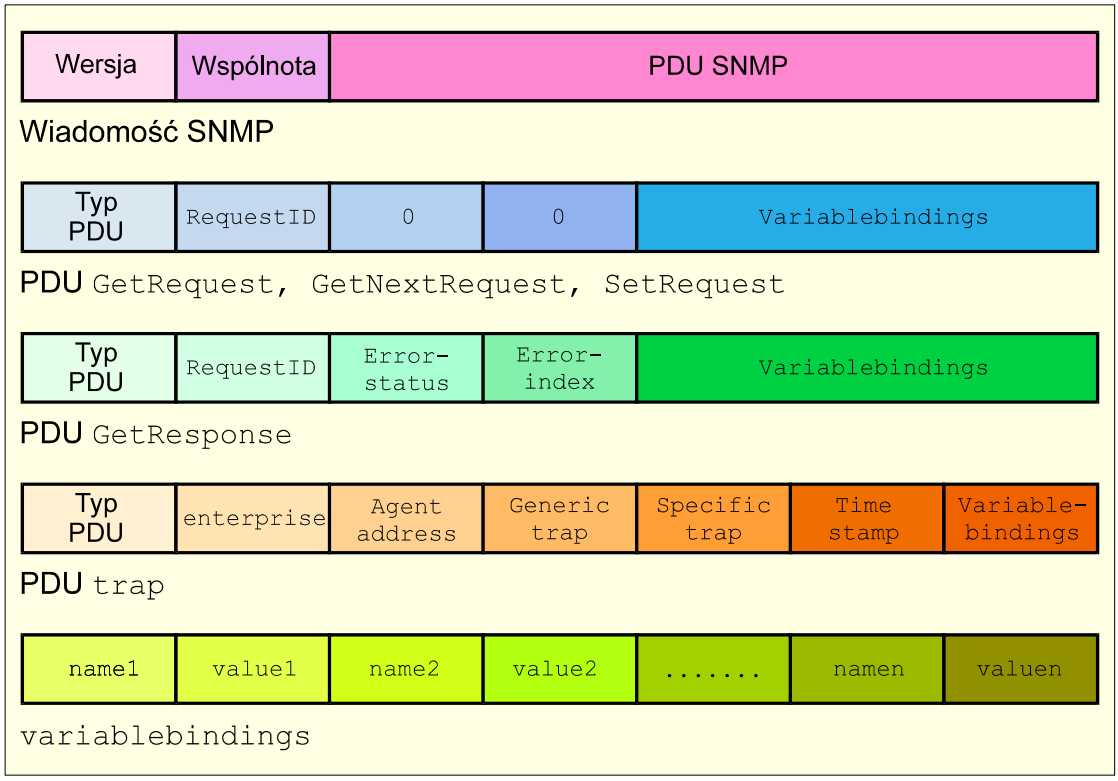
Elementem, który odpowiada za komunikowanie się oddalonych jednostek systemów zarządzania opartych na filozofii SNMP jest protokół zdefiniowany w RFC 1157.

Protokół umożliwia jedynie pobieranie oraz modyfikowanie wartości zmiennych, wykorzystując w tym celu następujące operacje podstawowe:

- `get`: pozwala stacji zarządzającej na pobranie wartości obiektu od agenta;
- `set`: powoduje ustawienie pożądanej wartości obiektu;
- `trap`: pozwala agentowi na samodzielne powiadomienie stacji zarządzającej o wystąpieniu określonego zdarzenia.

Nie jest natomiast możliwe dokonywanie zmian struktury bazy danych MIB poprzez dodawanie lub usuwanie jej składników (np. wierszy tablicy). Niedostępna jest także opcja wysyłania komend dotyczących aktualnie realizowanych akcji, zaś dostęp może dotyczyć jedynie obiektów stanowiących liście drzewa identyfikatorów. Wymienione ograniczenia redukują możliwości funkcjonalne systemu, ale równocześnie umożliwiają istotne uproszczenie implementacji SNMP

Wymiana danych pomiędzy stacją zarządzającą i agentem polega na przekazywaniu wiadomości. Każda wiadomość zawiera numer wersji SNMP, nazwę wspólnoty wymieniającej informacje oraz jeden z pięciu typów PDU.



Formaty danych SNMP

PDU typów `GetRequest, GetNextRequest` i `SetRequest` posiadają identyczny format jak `GetResponse` PDU, przenosząc pola `error-status` i `error-index` ustawione wartościami 0. Konwencja ta ogranicza ilość formatów wykorzystywanych przez SNMP do jednego.

Przeznaczenie pól informacyjnych

Pole	Opis
`version`	Wersja SNMP (RFC 1157 definiuje wersję 1)
`community`	Skojarzenie agenta SNMP ze zbiorem jednostek aplikacyjnych. Nazwa wspólnoty jest wykorzystywana do celów uwierzytelniania
`request- id`	Wyróżnik kolejnych żądań, które są obsługiwane równolegle
`error - status`	Wskazanie statusu przetwarzania żądania. Możliwe wartości: noError (0), tooBig (1), noSuchName (2), badValue (3), readOnly (4), genErr (5)
`error- index`	Przy różnym od zera statusie może dostarczać informacje o zmiennej z listy, która spowodowała wystąpienie błędu (zmienna stanowi reprezentację zarządzanego obiektu).
`variablebindings`	Lista nazw zmiennych oraz odpowiadających wartości. W niektórych przypadkach (np. `GetRequest PDU`) wartość pola wynosi `null`.
`Enterprise`	Typ obiektu, który wyzwolił pułapkę (bazuje na `sysObjectID`)
`agent-addr`	Adres obiektu, który wyzwolił pułapkę
`generic-trap`	Typ pułapki. Możliwe wartości: coldStart (0), warmStart (1), linkDown (2), linkUp (3), authentication-Failure (4), egpNeighborLoss (5), enterprise-Specific (6)
`specific -trap`	Kod pułapki typu `specific`
`time-stamp`	Czas pomiędzy (re)inicjalizacją jednostki i wyzwoleniem pułapki. Zawiera

	wartość `SysUpTime`.

W celu nadania wiadomości, jednostka SNMP realizuje następujące działania:

1. Konstruowanie PDU przy wykorzystaniu struktury ASN.1 zdefiniowanej w RFC 1157.
2. Przekazanie PDU funkcji uwierzytelnienia wraz z adresami źródłowym i przeznaczenia oraz nazwą wspólnoty SNMP. Funkcja bezpieczeństwa dokonuje właściwych przekształceń (szyfrowanie, dołączenie kodu uwierzytelniającego) i zwraca rezultat.
3. Jednostka protokołu zestawia wiadomość zawierającą pole wersji, nazwę wspólnoty oraz rezultaty realizacji fazy 2.
4. Przy wykorzystaniu BER *(Basic Encociing Rules)* zostaje stworzony nowy obiekt ASN.1, który otrzymuje funkcja transportowa.

W praktyce funkcja uwierzytelnienia nie jest wywoływana. Po odebraniu wiadomości, jednostka SNMP wdraża następujące akcje:

1. Dokonuje podstawowych sprawdzeń syntaktyki, eliminując wiadomości niepoprawne.
2. Weryfikuje numer wersji usuwając przekazy, których nie może poprawnie przetworzyć.
3. Przekazuje nazwę użytkownika, dostarczone w wiadomości PDU oraz adresy źródła i docelowy funkcji uwierzytelniającej. Jeśli wynik sprawdzenia jest negatywny komunikat zostaje anulowany z równoczesnym wyzwoleniu pułapki. W innym przypadku PDU w formie obiektu ASN.1 zostaje przekazana do dalszego przetwarzania.
4. Jednostki PDU niepoprawne syntaktycznie zostają skasowane, zaś poprawne są przetwarzane przy uwzględnieniu wymagań polityki dostępowej ustalonych na podstawie dostarczonej nazwy wspólnoty SNMP.

W praktyce uwierzytelnienie polega jedynie na stwierdzeniu akceptowalności odebranej nazwy wspólnoty.

Wszelkie operacje realizowane przez SNMP polegają wykorzystaniu dostępu do reprezentacji obiektów. Możliwość ta istnieje tylko w odniesieniu do liści drzewa MIB, a więc obiektów skalarnych. W niektórych przypadkach wykorzystywana jest opcja grupowania operacji tego samego typu (`get, set, trap`) w jednej wiadomości. Technika ta istotnie redukuje obciążenie zasobów systemu zarządzania.

1.1.1.10 Ograniczenia podstawowej wersji SNMP

Na podstawie analizy danych literaturowych możliwe jest wskazanie następujących potencjalnych niedogodności wynikających z wykorzystania SNMP:

1. SNMP może nie być wystarczający do zarządzania rzeczywiście dużymi sieciami. Wykorzystanie trybu odpytywania powoduje, że tylko ograniczona ilość elementów może być efektywnie obsługiwana.
2. SNMP nie posiada rozwiązań umożliwiających sprawne pozyskiwanie danych o dużej objętości (np. kompletnych tablic routingowych).
3. Agent generujący komunikat o wyzwoleniu pułapki przesyłany typowo z wykorzystaniem zestawu UDP/IP nie dysponuje żadnym mechanizmem ustalania, czy wiadomość dotarła do stacji zarządzającej.
4. Podstawowa wersja SNMP jest wyposażona jedynie w szczątkowe mechanizmy uwierzytelniające, co powoduje że bezpieczniej jest używać go jedynie do monitoringu, a nie sterowania.
5. SNMP nie obsługuje poleceń nakazowych - jedyną możliwą techniką oddziaływania na zasoby agenta jest ustawianie wartości obiektów. Rozwiązanie to jest mniej efektywne niż wywoływanie zdalnych procedur z parametrami, warunkami, statusem i raportowaniem rezultatów.
6. Model MIB SNMP jest uproszczony i w ograniczonym stopniu może być wykorzystany przez aplikacje, które generują złożone zapytania bazujące na typach lub wartościach obiektów.
7. SNMP nie realizuje komunikacji w relacji zarządca - zarządca. W szczególności brak mechanizmu

rozpoznawania zasobów nadzorowanych przez inne stacje zarządzające.

Uwzględniając te i inne niedoskonałości pierwotnej wersji SNMP, zadecydowano o stworzeniu jego kolejnej wersji, znanej obecnie jako SNMPv2.

1.1.2 Koncepcyjne podstawy SNMPv2

1.1.2.1 Wprowadzenie

Protokół SNMP w swojej podstawowej formie został pomyślany jako zminimalizowany zestaw procedur umożliwiający realizację uproszczonego zarządzania siecią. Po roku 1988 stało się jasne, że zarządzanie powinno rozpatrywane w dwóch podstawowych horyzontach czasowych: obecnym, odzwierciedlającym bieżący poziom rozbudowy sieci oraz perspektywicznym, zdolnym do sprostania wymogom o wiele bardziej rozbudowanych środowisk systemowych.

W konsekwencji, za celowe uznano wdrożenie dla celów doraźnych implementacji SNMP, podczas gdy realizację zadań długofalowych przejąć miały rozwiązania bazujące na systemowym podejściu OSI. Implementacją realizującą filozofię OSI w środowiskach sieci internetowych stał się z czasem CMOT *(CMIP over TCP/IP)*. Z kilku powodów strategii dwutorowego wdrażania systemów zarządzania nie udało się w pełni zrealizować:

1. Pierwotnie zakładano, że SMI i MIB protokołu SNMP stanowić będą podzbiory analogicznych rozwiązań przeznaczonych dla środowiska OSI, co jak planowano, umożliwić miało szybkie oraz relatywnie łatwe przejście na kolejny etap rozwojowy. Jednakże zorientowane obiektowo podejście OSI nie dało się pogodzić z wypracowanym podczas pośpiesznego wdrażania SNMP i w konsekwencji postulowana więź nie została ustanowiona. Rezultatem są duże trudności z zastępowaniem SNMP przez rozwiązania OSI.
2. Tworzenie stabilnych standardów OSI oraz związana z tym dostępność realizujących je produktów trwały dłużej niż pierwotnie zakładano. W ten sposób otworzyły się możliwości zaistnienia na rynku rozwiązań mniej złożonych, ale za to kompletnych. SNMP wykorzystał tą szansę i obecnie jest szeroko rozpowszechniony.

Panuje ogólna zgodność co do faktu, że w wielu dużych i złożonych systemach telekomunikacyjnych, które stają się obecnie coraz bardziej liczne, SNMP nie jest w stanie spełnić wymagań użytkowników. Istnieje jednak równocześnie bardzo wiele instalacji sieciowych, w których jego wykorzystanie jest w pełni racjonalne.

Podstawowym czynnikiem, który ograniczał zainteresowanie SNMP w wersji podstawowej był brak kompleksowej implementacji mechanizmów bezpieczeństwa. W efekcie systemy funkcjonujące w oparciu o tą filozofię nie posiadały zabezpieczeń przed nieuprawnioną modyfikacją konfiguracji sieciowej.

Problem ten próbowano rozwiązać, publikując w lipcu 1992 r zestaw RFC znany jako *„Secure SNMP”*. Propozycja ta nie pozwalała jednak na poprawę innych właściwości użytkowych oraz funkcjonalnych i dlatego niemal równocześnie pojawiło się nowe rozwiązanie, określane mianem SMP *(Simple Management Protocol)*.

Normatywy SMP stanowił zestaw ośmiu dokumentów nie stanowiących jednak RFC, ponieważ była to propozycja „prywatna”. Rozszerzenia wprowadzane przez SMP dotyczyły następujących kategorii:

- ***Przedmiotu*** - SMP został zaprojektowany jako aplikacja wspomagająca zarządzanie zasobami rozumianymi szerzej niż tylko „sieciowe”. W efekcie stawało się możliwe również zarządzanie aplikacjami i całym systemem, a to dzięki uwzględnieniu mechanizmów wzajemnego komunikowania się stacji zarządzających.
- ***Szybkości działania i efektywności*** - SMP pozostawał „prosty” by umożliwić małe i szybkie implementacje Główną zmianę stanowił natomiast transfer zbiorczy *(bulk transfer)*,

pozwalający na efektywną wymianę danych o dużej objętości.

- ***Bezpieczeństwa*** - SMP zaadaptował do swoich potrzeb rozwiązania proponowane przez *Secure SNMP*.
- ***Kompatybilności*** - uwzględniono możliwość korzystania z mechanizmów TCP/IP, OSI oraz innych architektur sieciowych, przewidziano możliwość współpracy z platformami SNMP.

Publikacja Secure SNMP oraz SMP wyzwoliła burzliwą dyskusję licznych środowisk internetowych zainteresowanych szybkim przejściem z bazowej wersji SNMP na implementację pozbawioną znaczących niedogodności. Rezultatem było wypracowanie poglądu, że prace rozwojowe powinny być prowadzone w oparciu o propozycję SMP, zaś nową wersję protokołu zarządzania nazwano SNMPv2. W celu odróżnienia, dotychczasowy standard stał się więc SNMPv1.

Prace rozwojowe realizowano w dwóch grupach roboczych, z których pierwsza analizowała aspekty bezpieczeństwa, druga zaś aktualizowała protokół oraz format informacji zarządzania. Opracowanie zawierające rezultaty prac obydwu zespołów opublikowano w formie propozycji standardów internetowych w marcu 1993 r.

Po kilku latach doświadczeń z SNMPv2, IETF zadecydował, że konieczne są pewne uzupełnienia. Aktualizację standardu połączono z eliminacją dotychczasowych funkcji bezpieczeństwa, dodatkowo zaś SNMPv2 przejął format kontenera wiadomości v1 wraz z właściwą mu koncepcją wspólnoty SNMP. Rezultat przeprowadzonych działań stał się dostępny jako *„community-based SNMPv2”* lub w skrócie SNMPv2C.

Eliminację funkcji bezpieczeństwa z poprawionej wersji SNMP ocenić trzeba negatywnie, aczkolwiek jej realizacja nastąpiła z kilku ważnych powodów. Wytwórcy oraz użytkownicy przyjęli mianowicie rozwiązania zaproponowane w 1993 bez entuzjazmu, wskazując na jego liczne niedogodności. Kiedy rozpoczęto modyfikację standardu, wydawało się, że poprawę sytuacji można uzyskać dokonując kosmetycznych zmian dotychczasowej wersji, ale nadzieje te nie miały oparcia w faktach.

1.1.2.2 Podstawowe rozszerzenia

SNMPv2 może być wykorzystywany do realizacji zarówno scentralizowanej, jak i rozproszonej strategii zarządzania. W ostatnim przypadku, niektóre jednostki mogą funkcjonować równocześnie jako zarządca oraz agent. Oznacza to, że akceptowane są komendy z elementu nadrzędnego, które powodują bądź udostępnianie informacji przechowywanej w stacji pośredniego szczebla, bądź zbiorczych danych na temat zbioru podporządkowanych jej agentów. Stacje pośrednie mogą również w stosunku do nadrzędnych funkcjonować w trybie wyzwalania pułapek.

Rozszerzenia wprowadzone przez SNMPv2 dotyczą następujących kategorii:

- struktury informacji zarządzania (SMI);
- wymiany danych pomiędzy stacjami zarządzającymi;
- specyficznych operacji protokołu.

SMI została rozszerzona na kilka sposobów: makrodefinicja typów obiektowych obejmuje nowe typy danych, wszystkie one zostały znacznie lepiej udokumentowane. Ponadto wprowadzono nową konwencję tworzenia oraz usuwania wierszy tablic, która funkcjonując w sposób zbliżony do wykorzystywanego w RMON MIB.

Wykorzystywana przez SNMPv2 baza danych MIB zawiera podstawowe informacje ruchowe dotyczące operacji realizowanych przez protokół, co stanowi analogię do funkcjonowania grupy `snmp` w MIB-II. Baza obejmuje także dane konfiguracyjne aplikacji zarządzającej albo agenta.

Najbardziej istotne zmiany dotyczące operacji protokołu polegają na wprowadzeniu dwóch nowych jednostek PDU: `GetBulkRequest` pozwala na efektywne pozyskiwanie danych o dużej objętości, natomiast `InformRequest` wykorzystywana jest do transferowania informacji typu `Trap` pomiędzy stacjami zarządzającymi.

1.1.2.3 Protokół wymiany danych SNMPv2

Jednostki danych protokołu SNMPv2 są, podobnie jak w poprzedniej wersji, przekazywane w ramach struktur znanych jako wiadomości. Struktury te zawierają elementy przeznaczone do realizacji funkcji bezpieczeństwa. Oznacza to, że format oraz znaczenie nagłówków wiadomości wyznacza przyjęty schemat administracyjny, stanowiący również politykę uwierzytelniania oraz zapewniania poufności.

SNMPv2 oferuje trzy typy dostępu do informacji zarządzania:

- *Zarządca-agent; żądanie-odpowiedź* - wykorzystywany do pozyskiwania lub modyfikowania informacji skojarzonych z elementem zarządzanym.
- *Agent-zarządca; bez potwierdzania* - przeznaczony do powiadamiania jednostek zarządzających o wystąpieniu zdarzeń, które prowadzą do zmian informacji skojarzonej z elementem zarządzanym.
- *Zarządca-zarządca; żądanie-odpowiedź* - tryb wzajemnego porozumiewania się jednostek zarządzających, przeznaczony do powiadamiania jednej ze stron o informacji zarządzania skojarzonej z drugim elementem.

Pierwsze dwa typy interakcji istnieją również w bazowej wersji SNMP. Jedynie ostatnia jest właściwa tylko SNMPv2.

Jednostki PDU protokołu SNMPv2 są przekazywane w ramach wiadomości zawierających również nazwę wspólnoty SNMP wykorzystywaną dla celów uwierzytelnienia. Wszystkie przedstawione dotychczas informacje, dotyczące nazw i profilów wspólnoty oraz polityki sterowania dostępem, stosują się również do SNMPv2. W tym przypadku pole nagłówkowe `version` zawiera wartość 1 (SNMPv1 stanowi wersję 0). Wykorzystanie formatu wiadomości SNMPv1 jako zewnętrznej ramki jednostek PDU v2 jest określane mianem SNMPv2 bazującej na pojęciu wspólnoty (*community-based*) lub SNMPv2C.

W celu przekazania jednostki PDU element SNMPv2 realizuje następujące operacje:

1. Zestawienie PDU przy wykorzystaniu struktury ASN.1 zdefiniowanej w specyfikacji protokołu.
2. Przekazanie PDU do funkcji uwierzytelniającej wraz z adresami źródła i punktu przeznaczenia oraz nazwą wspólnoty. Funkcja ta dokonuje stosownych przekształceń (szyfrowanie, dołączenie kodu uwierzytelniającego) i zwraca rezultat.
3. Utworzenie wiadomości zawierającej pole wersji, nazwę wspólnoty oraz rezultaty realizacji kroku 2.
4. Otrzymany obiekt ASN.1 zostaje zakodowany przy wykorzystaniu bazowych reguł BER i przekazany usłudze transportowej.

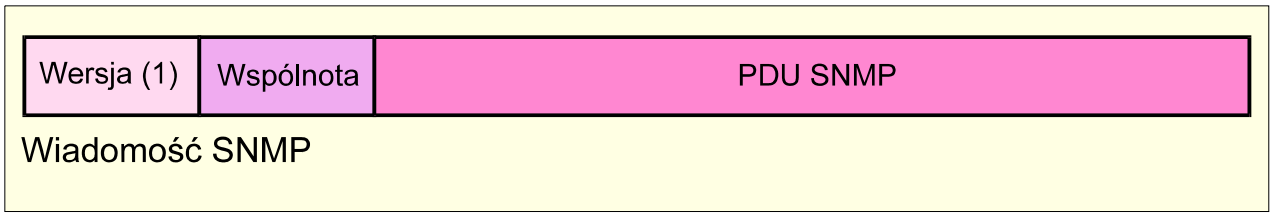
W praktyce uwierzytelnienie (pkt. 2) nie jest realizowane. Po odebraniu wiadomości wdrażany jest następujący algorytm jej przetwarzania:

1. Sprawdzenie poprawności syntaktycznej i usuwanie wiadomości niepoprawnych składniowo;
2. Weryfikacja numeru wersji i eliminacja jednostek niewłaściwie oznaczonych;
3. Przekazanie nazwy użytkownika, otrzymanych PDU oraz adresów źródła i punktu przeznaczenia dostarczonych przez usługę transportową do funkcji uwierzytelniania. Jeśli sprawdzenie wypada niepomyślnie jednostka protokołu anuluje wiadomość i generuje pułapkę, w przeciwnym przypadku PDU zostaje zwrócona w formie obiektu ASN.1.
4. Sprawdzenie poprawności syntaktycznej i usuwanie PDU niepoprawnych składniowo. Jednostki poprawne zostają przetworzone stosownie do polityki zdefiniowanej dla wskazanej przez wiadomość wspólnoty.

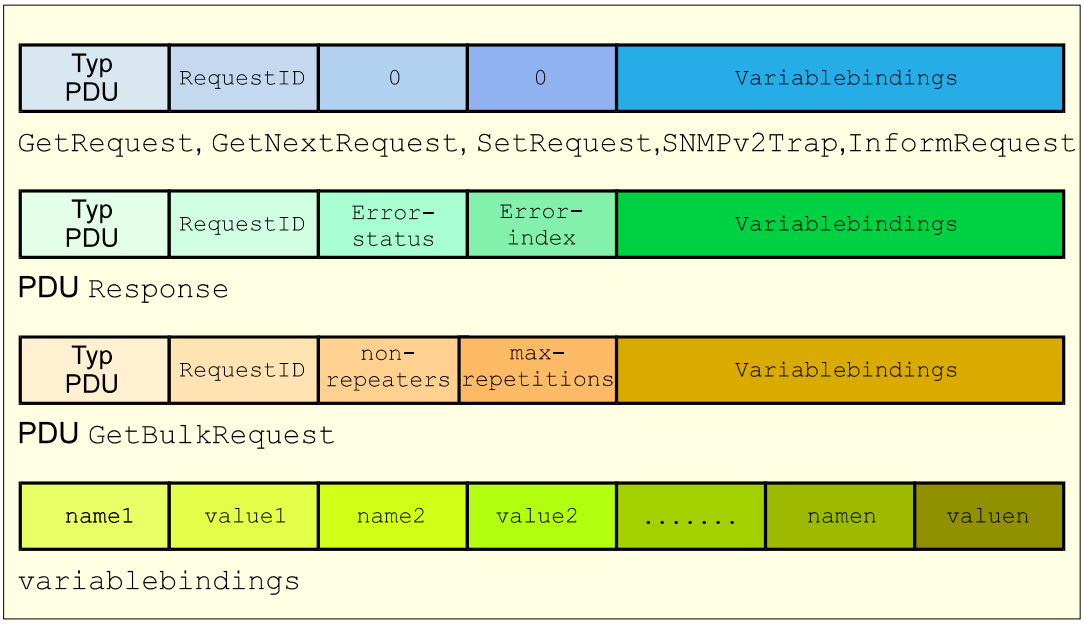
Relacje pomiędzy wartością `MAX-ACCESS` i trybem dostępu.

Wartość MAX-ACCESS	Tryb dostępu	
	READ-ONLY	READ-WRITE
read-only	Możliwe operacje get i trap	
read-write	Możliwe operacje get i trap	Możliwe operacje get, set i trap
read-create	Możliwe operacje get i trap	Możliwe operacje get, set, create i trap
accessible-for-notify	Możliwa operacja trap	
not-accessible	Brak dostępnych operacji	

W praktyce funkcja uwierzytelnienia sprawdza jedynie, czy odebrana nazwa wspólnoty jest odpowiednia dla jednostki SNMPv2, która wygenerowała wiadomość.



Struktura PDU SNMPv2



Formaty jednostek protokołu

Jednostki typu GetRequest, GetNextRequest, SetRequest oraz Trap posiadają taki sam format jak PDU Response i InformRequest, z polami error-status i error-index ustawionymi wartościami 0. Wykorzystanie jednolitego formatu wiadomości pozwala na znaczne uproszczenie implementacji elementów systemów zarządzania.

Porównanie PDU wersji 1 i 2 protokołu SNMP

SNMPv1	SNMPv2	Kierunek	Opis
GetRequest	GetRequest	Zarządca - agent	Żądanie wartości obiektu
GetNextRequest	GetNextRequest	Zarządca - agent	Żądanie następnej wartości

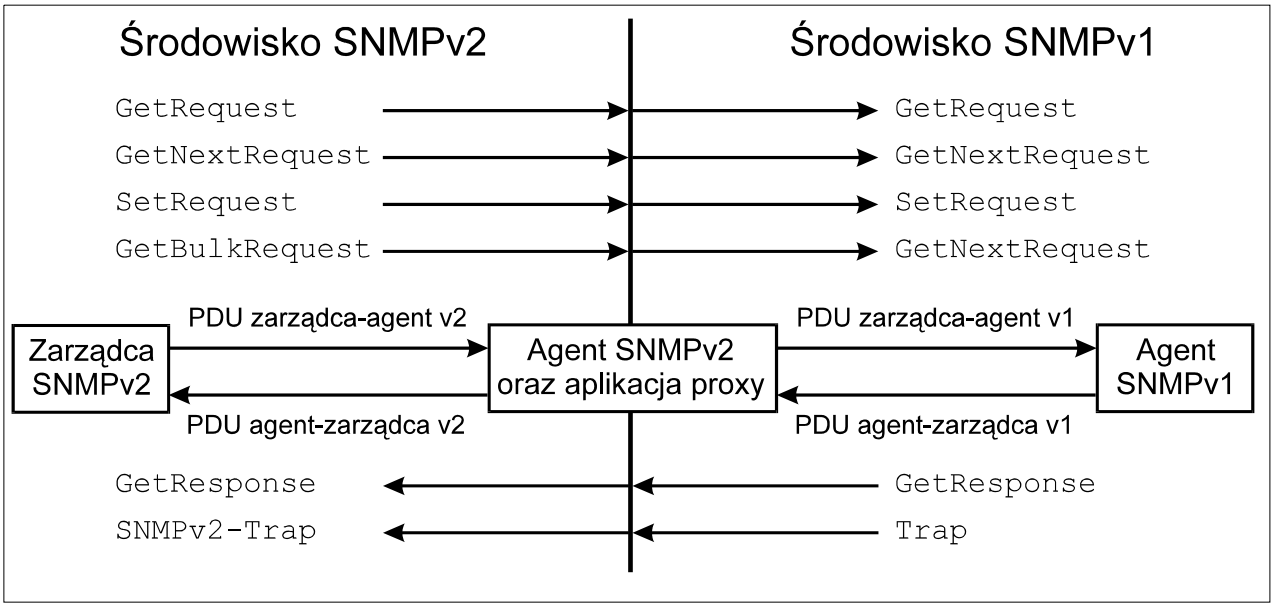
-	GetBulkRequest	Zarządca - agent	Żądanie wielu wartości
SetRequest	SetRequest	Zarządca - agent	Ustawienie wartości
-	InformRequest	Zarządca - zarządca	Przekaz bez wywołania
GetResponse	Response	Agent - zarządca lub zarządca - zarządca	Odpowiedź na żądanie zarządcy
Trap	SNMPv2-Trap	Agent - zarządca	Przekaz bez wywołania

Zestawienie pól występujących we wszystkich jednostkach PDU SNMPv2 obejmuje następujące elementy:

- request-id - wartość tego pola w PDU stanowiącej odpowiedź musi być identyczna jak w powodującym ją żądaniu, co pozwala zarządcy na jednoznaczne ich przyporządkowanie podczas operacji wielokrotnych.
- error-status - wartość różna od zera wskazuje, że podczas przetwarzania żądania wystąpił wyjątek.
- error-index - jeśli pole error-status zawiera wartość różną od zera, pole error-index wskazuje obiekt listy variable-bindings, który spowodował wystąpienie błędu. Pierwsza wartość listy posiada indeks 1, druga - 2 itd.
- variable-bindings - pole umożliwia jednoczesne wywołanie operacji w stosunku do grupy reprezentacji obiektów, określonej sekwencją par wartości, z których pierwsza jest identyfikatorem, druga zaś elementem następującej listy:
  - wartość - stan konkretnego obiektu wskazanego w PDU typu request;
  - unSpecified - w żądaniu dostarczenia zawartości występuje wartość NULL;
  - noSuchObject - wskazanie, że agent nie obsługuje obiektu wskazanego w żądaniu;
  - noSuchInstance - wskazanie, że żądana reprezentacja nie istnieje;
  - endOfMibView - wskazanie, że podjęto próbę dostępu do obiektu o identyfikatorze wykraczającym poza zakres obsługiwany przez MIB agenta.

Nowa wersja protokołu SNMP jest podobna do poprzedniej w części wykorzystującej już istniejące formaty PDU. Istotne zmiany wprowadziło natomiast zdefiniowanie operacji GetBulkRequest i InformRequest, a także modyfikacja trybu wykonywania opcji get poprzez rezygnację z jej niepodzielności. W efekcie pełna interoperacyjność jest możliwa pod warunkiem modyfikacji trybu funkcjonowania agenta proxy oraz zdefiniowaniu uniwersalnego algorytmu działania stacji zarządzającej.

Najprostszym rozwiązaniem ewolucyjnego przechodzenia na nową wersję protokołu jest pozostawienie istniejących agentów starszej wersji i komunikowanie się z nimi za pośrednictwem agentów proxy. Ich funkcje mogą realizować odpowiednio skonfigurowane aplikacje agentów SNMPv2, które pośredniczą w wymianie prowadzonej za pośrednictwem obydwu wersji protokołu.



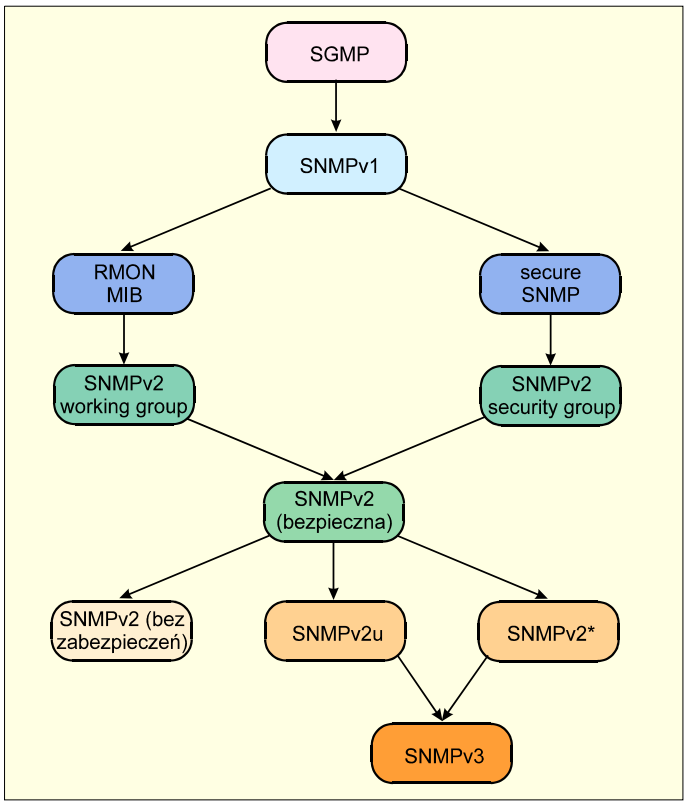
Funkcjonowanie agenta proxy

Jednostki PDU SNMPv2 przekazywane przez stację zarządzającą są przekształcane do postaci akceptowalnej przez agenta SNMPv1.

1.1.3 Architektura i aplikacje SNMPv3

1.1.3.1 Wprowadzenie

W swojej ostatecznej formie, SNMPv2 nie posiada prawie żadnych mechanizmów gwarantowania bezpieczeństwa. W celu eliminacji tej istotnej niedogodności wprowadzono do użytku kolejne rozszerzenie, znane jako SNMPv3. Określenie nowej wersji mianem „rozszerzenia" jest jak najbardziej uzasadnione, ponieważ jej specyfikacja obejmuje jedynie funkcje związane z aspektem bezpieczeństwa, odwołując się w pozostałych obszarach do uregulowań zdefiniowanych we wcześniejszych wersjach protokołu SNMP.



Etapy rozwojowe SNMP

W styczniu 1998 roku opublikowano serię proponowanych standardów internetowych, znanych odtąd jako RFC 2271 - 2275. Wymienione dokumenty stanowią ramowy schemat „doposażenia" funkcji oferowanych przez obydwie wcześniejsze wersje protokołu o pojmowane w sposób nowoczesny mechanizmy bezpieczeństwa. W rezultacie implementacja określana mianem SNMPv3 obejmuje wszystkie aspekty wyspecyfikowane w RFC 2271 - 2275, wykorzystując formaty PDU oraz inne funkcje zdefiniowane zapisami SNMPv2 (która bazuje z kolei na SNMPv1). Wstępna klauzula standardu określa jego charakter stwierdzeniem: „SNMPv3 stanowi SNMPv2 plus bezpieczeństwo i administracja".

1.1.3.2 Ogólna charakterystyka SNMPv3

Uogólnioną architekturę SNMP definiują zapisy RFC 2271. Architektura ta stanowi w istocie modularny opis elementów funkcjonalnych i bezpieczeństwa, które są wykorzystywane przez implementacje stacji zarządzającej albo agenta. Prace rozwojowe dotyczące architektury SNMP prowadzono uwzględniając następujące uwarunkowania:

1. Potrzebę wykorzystania tych spośród rezultatów dotychczasowych prac, co do których istniały znaczące doświadczenia implementacyjne oraz zgodność co do ich dużego znaczenia. W efekcie architekturę SNMP oraz realizację funkcji bezpieczeństwa SNMPv3 oparto w znacznym stopniu na mechanizmach udostępnianych przez SNMPv2.
2. Potrzebę możliwości bezpiecznej realizacji żądań klasy `Set` we wszelkich strukturach sieciowych, w tym dostępnych publicznie.
3. Niezbędność modularyzacji architektury systemu, która:
   - pozwoli na implementowanie mechanizmów zarządzania w środowiskach o silnie zróżnicowanych właściwościach (od małych sieci korporacyjnych, po duże systemy publiczne);
   - umożliwi ciągły postęp na ścieżce standaryzacyjnej, nawet bez kompleksowego porozumienia wszystkich zainteresowanych środowisk;
   - będzie w stanie realizować alternatywne modele bezpieczeństwa.

4. Konieczność zachowania małej komplikacji SNMP w stopniu, w którym jest to możliwe.

Wykorzystując założenia, sformułowano szereg wytycznych obowiązujących podczas formułowania specyfikacji SNMPv3. Lista najważniejszych spośród nich obejmuje kolejno:

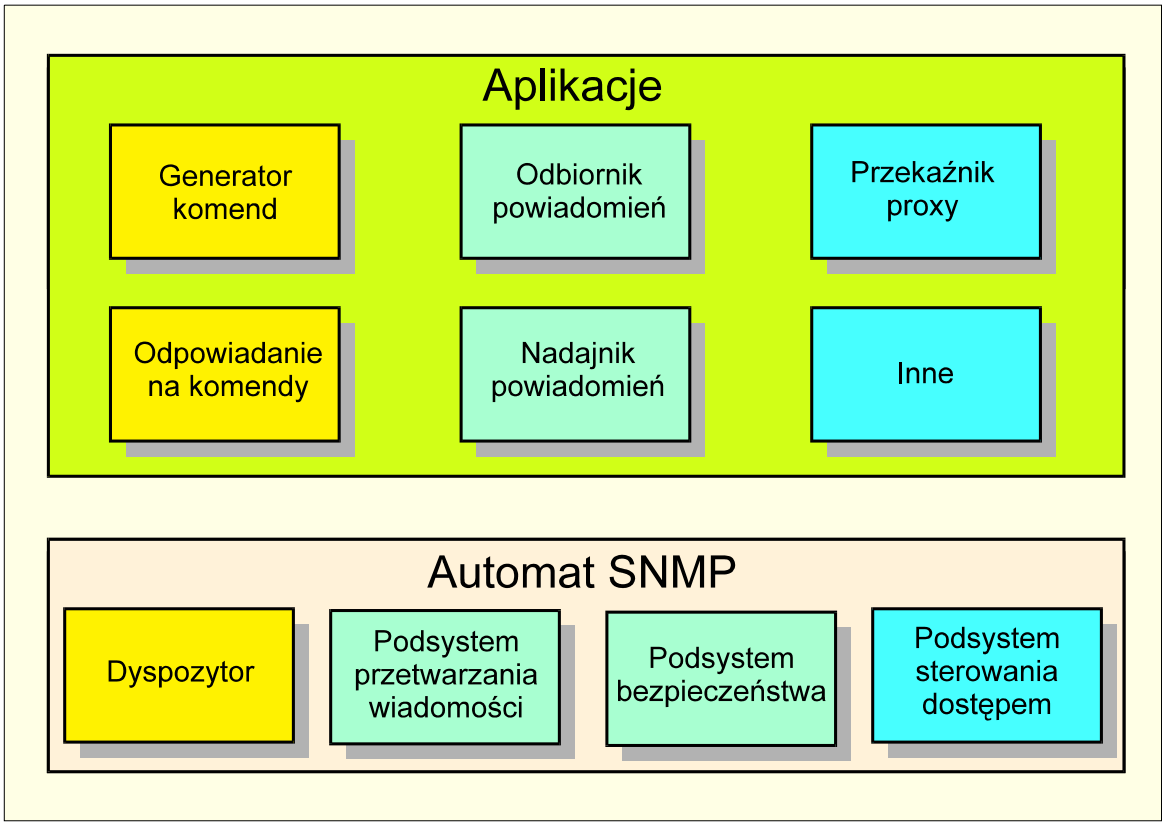
1. Dążenie do uniwersalizmu architektury: Powinna ona zostać zdefiniowana w sposób identyfikujący koncepcyjne rozgraniczenia pomiędzy poszczególnymi dokumentami. Podsystemy muszą opisywać abstrakcyjne usługi świadczone przez określone fragmenty modelu ramowego SNMP, zaś aplikacjom koncepcyjnych podsystemów powinny odpowiadać abstrakcyjne interfejsy usługowe.
2. Ekstremalna zwartość dokumentacji: elementy procedur oraz obiekty MIB niezbędne do przetwarzania dowolnego fragmentu ramowego modelu SNMP powinny definiować zasadniczo te same dokumenty. W efekcie, zmiany wprowadzone w określonych modułach MIB nie powinny wpływać na pozostałą część implementacji, zaś same dokumenty muszą być użyteczne w różnych etapach kompleksowego procesu standaryzacji.
3. Zdalna konfigurowalność: Podsystemy bezpieczeństwa oraz sterowania dostępem wprowadzają obszerny zestaw parametrów konfiguracyjnych SNMP, w tym wymagających częstych zmian dla utrzymania wysokiej odporności systemu na ataki. Możliwość funkcjonowania dużych systemów zarządzania jest zatem uwarunkowana dostępnością zdalnych metod konfiguracyjnych.
4. Sterowalna kompleksowość: Podczas, gdy procedury prostych elementów zarządzania powinny wykorzystywać minimalny zestaw zasobów systemowych, bardziej rozbudowane implementacje muszą za cenę zwiększonego zapotrzebowania oferować o wiele szerszy zakres realizowanych funkcji. Konkretne rozwiązania muszą elastycznie godzić wymienione sprzeczności, w sposób, w którym duże środowiska stanowią logiczne rozszerzenia mniejszych.
5. Odporność na ataki: modele oraz podsystemy bezpieczeństwa MUSZĄ chronić przynajmniej przed atakami następujących typów: modyfikacją informacji, maskaradą, ingerencją w strumień wiadomości oraz wykluczaniem. Nie jest natomiast wymagana odporność na odmowę obsługi oraz analizę ruchu.

Jakkolwiek nie wszystkie wymienione wytyczne udało się zrealizować w całości, jest niezaprzeczalnym faktem, że samo ich sformułowanie przyczyniło się w znacznym stopniu do poprawy własności użytkowych SNMP.

1.1.3.3 Architektura SNMP

Zgodnie z zapisami RFC 2271, architekturę SNMP stanowi zestaw współpracujących w sposób rozproszony jednostek SNMP. Każda z nich, funkcjonując jako agent zarządca lub w obydwu rolach równocześnie, implementuje określony fragment funkcjonalności SNMP.

Świadczenie usług realizowane jest w warunkach współdziałania modułów tworzących daną jednostkę SNMP. Wykorzystywane w tym celu interakcje mogą być modelowane jako zbiór abstrakcyjnych prymitywów oraz zestaw ich parametrów. W dalszym ciągu zaprezentowana zostanie zarówno wewnętrzna struktura jednostki SNMP, jak i usługi jej modułów.



Jednostka SNMP

W skład każdej jednostki wchodzi automat SNMP *(SNMP engine)*, który realizuje odbiór i nadawanie wiadomości, uwierzytelnianie oraz funkcje kryptograficzne, a także nadzoruje dostęp do zarządzanych obiektów. Rola wypełniana przez daną jednostkę jest wyznaczana każdorazowo zestawem tworzących ją modułów. Jeden zbiór odpowiada aplikacji agenta, częściowo inny pozwala funkcjonować w postaci zarządcy. Modułowa struktura konfiguracji prowadzi wprost do możliwości definiowania różnych wersji danego modułu. Pozwala to miedzy innymi na:

- specyfikowanie alternatywnych albo rozszerzonych funkcji dotyczących wybranego aspektu SNMP bez potrzeby definiowania od podstaw nowego standardu;
- wyznaczanie przejrzystych strategii współistnienia oraz ewolucji.

Uproszczone definicje modułów architektury SNMP zdefiniowanych zapisami RFC 2271 i 2273 posiadają następującą postać:

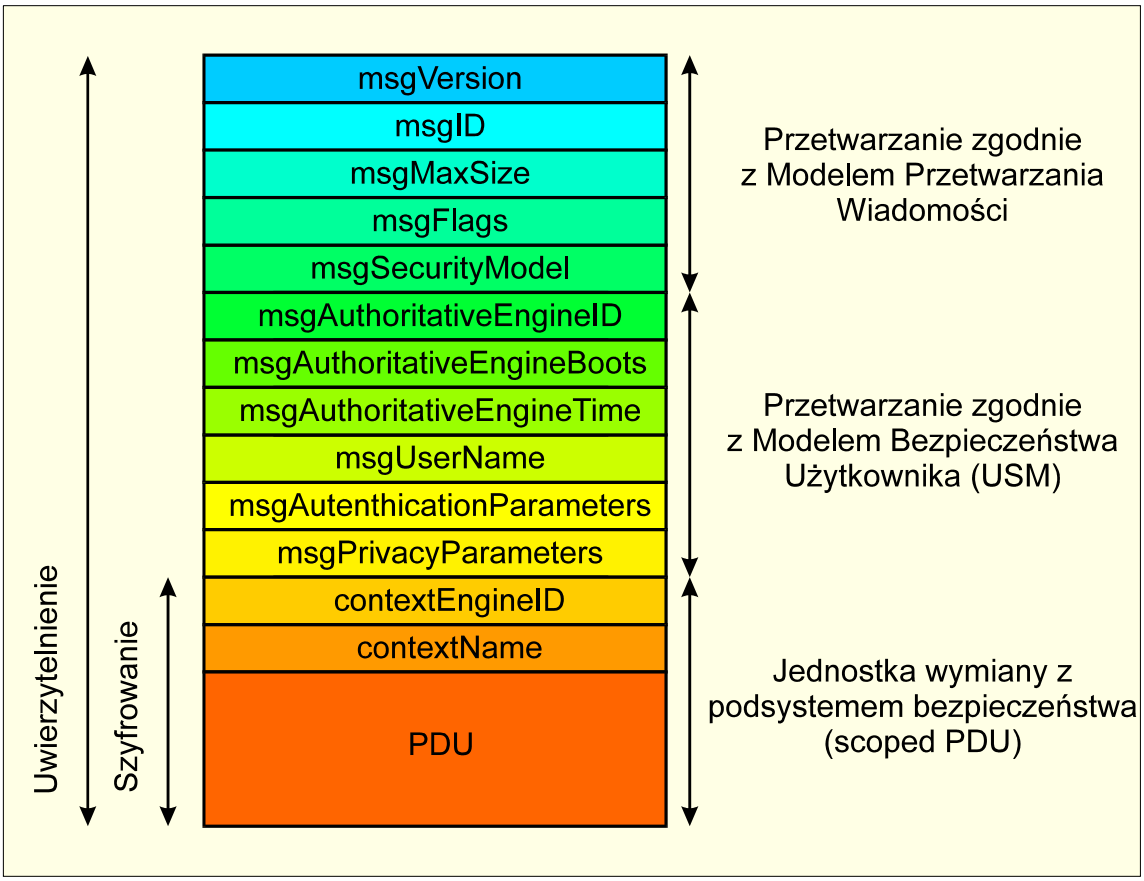
- Dyspozytor *(dispatcher)* - umożliwia automatowi równoległą obsługę wielu wersji SNMP. Odpowiada za: wymianę PDU na stykach z aplikacją i podsystemem przetwarzania wiadomości oraz ich nadawanie i odbiór za pośrednictwem sieci.
- Podsystem Przetwarzania Wiadomości *(Message Processing Subsystem)* - realizuje przygotowanie wiadomości do nadania oraz ekstrakcję danych z wiadomości odebranych.
- Podsystem Bezpieczeństwa *(Security Subsystem)* - świadczy usługi związane z zabezpieczaniem informacji (uwierzytelnianie, poufność itp.).
- Podsystem Sterowania Dostępem *(Access Control Subsystem)* - dostarcza aplikacji zestawu usług uwierzytelniających wykorzystywanych do weryfikacji praw dostępu.
- Generator Komend *(Command Generator)* - inicjuje operacje *Get, GetNext, GetBulk* i *Set* oraz przetwarza otrzymane odpowiedzi.
- Odbiornik Komend *(Command Responder)* - odbiera PDU adresowane do lokalnego automatu SNMP, następnie wykonuje odpowiednie operacje w warunkach nadzorowania praw dostępu i generuje wymagane odpowiedzi.
- Nadajnik Powiadomień *(Notification Originator)* - monitoruje system wykrywając

wystąpienie określonych zdarzeń lub warunków i generuje wiadomości `Trap` oraz `Inform`.

- Odbiornik Powiadomień *(Notification Receiver)* - oczekuje na powiadomienia oraz generuje odpowiedzi na wiadomości zawierające PDU typu `Inform`.
- Przekaźnik proxy *(Proxy Forwarder)* - przekazuje wiadomości wymieniane przez jednostki funkcjonalne SNMP korzystające z jego pośrednictwa.

1.1.3.4 Przetwarzanie wiadomości

Sposób przetwarzania wiadomości SNMP specyfikuje RFC 2272, w którym określono podstawowe zadania mechanizmów systemowych, a w tym: wymianę PDU z modułem sterownika ruchu, ich wstawianie (ekstrakcję) do(z) wiadomości oraz obsługę pól nagłówka dedykowanych funkcjom bezpieczeństwa. Strukturę wiadomości SNMP ilustruje rys. C.10.



Rys. C.10. Struktura wiadomości SNMPv3

Zestaw pól przenoszonych przez wiadomości SNMPv3 obejmuje:

msgVersion - ustawione wartością snmpv3(3).

msgID - unikalny identyfikator wykorzystywany przez komunikujące się jednostki SNMP do koordynowania transferu wiadomości typów *request* i *response* oraz przez procesor wiadomości w trakcie wymiany danych z różnymi składnikami architektury systemu. Zakres wartości pola wynosi od 0 do 2^{31}-1.

msgMaxSize: - przenosi maksymalną długość wiadomości możliwej do zaakceptowania i przetwarzania przez nadawcę (zakres od 484 do 2^{31}-1).

msgFlags - oktet, którego trzy najmniej znaczące bity stanowią zespół wskaźników: *reportableFlag, privFlag, authFlag*. Jeśli flaga *reportableFlag* = 1, to PDU typu *Report* musi zostać przekazana, jeśli wystąpi powód do jej generacji, natomiast przy wyzerowaniu flagi nadanie PDU nie jest

obowiązkowe. Nadajnik ustawia pole *reportableFlag* wartością „1" w wiadomościach z PDU *Get, Set* oraz *Inform*, zaś przekazywanie PDU *Response, Trap* lub *Report* oznaczane jest wyzerowaniem pola. Flaga ma charakter pomocniczy, bowiem umożliwia wskazanie konieczności nadania PDU typu *Report* pomimo, że odbiornik nie dysponuje właściwym kluczem deszyfrującym, umożliwiającym dostęp do zasadniczej zawartości komunikatu. Pola *privFlag* i *authFlag* wskazują funkcje bezpieczeństwa wykorzystane przez nadawcę wiadomości. Jeśli *privFlag* = 1, użyto szyfrowania, gdy *privFlag* = 0, wymagane jest uwierzytelnienie. Niedopuszczalna jest kombinacja *privFlag* = 1 i *authFlag* = 0.

MsgSecurityModel - identyfikator o wartości z zakresu od 0 do 2^{31}-1, wskazujący model bezpieczeństwa wykorzystany podczas generacji wiadomości. Dotychczas zarezerwowano wartości 1 (SNMPv1), 2 (SNMPv2c) i 3 (USM SNMPv3).

msgSecurityParameters - ciąg oktetów zawierający parametry generowane przez Podsystem Bezpieczeństwa nadawcy i przetwarzane po stronie odbiorcy. Zawartość pola nie podlega interpretacji przez Podsystem Przetwarzania Wiadomości i Dyspozytora.

contextEngineID - ciąg oktetów identyfikujący jednoznacznie jednostkę SNMP. Po stronie odbiorczej interpretacja pola dostarcza informacji, która z aplikacji powinna przetwarzać zawartość `scopedPDU`. Obowiązek ustawienia wartości pola w nadajniku spoczywa na aplikacji żądającej przesłania wiadomości.

contextName - ciąg oktetów identyfikujący właściwy kontekst z obsługiwanych przez skojarzony automat kontekstowy.

data - jednostka PDU, która zgodnie ze specyfikacją modelu przetwarzania wiadomości SNMPv3 musi spełniać wymogi standardu SNMPv2.

Pola od drugiego do piątego są określone w definicji ASN.1 jako `msgGlobalData`, ponieważ przenoszą informacje wykorzystywane przez Podsystem Przetwarzania do koordynacji obsługi i przetwarzania wiadomości. Odpowiednio trzy ostatnie elementy tworzą sekcję `msgData` stanowiąc typ `scopedPduData`, przy czym PDU zawiera informacje w kontekście określonym jednoznacznie zawartością pól `contextEngineID` oraz `contextName`. Informacje przenoszone w tym bloku wykorzystuje aplikacja przetwarzająca wiadomość.

1.1.3.5 Model bezpieczeństwa

Zorientowany na użytkownika model bezpieczeństwa systemu SNMPv3 *(User Security Model - USM)* definiuje formalnie dokument RFC 2274. Specyfikacja ta uwzględnia następujące aspekty:

- Uwierzytelnianie - zapewnia gwarancje integralności oraz pochodzenia danych przy wykorzystaniu kodu HMAC utworzonego przy użyciu funkcji MD5 lub SHA-1.
- Nadzór parametrów czasowych - chroni przed atakami polegającymi na opóźnianiu lub powtarzaniu wiadomości.
- Poufności - zabezpiecza zawartość wiadomości przed ujawnieniem dzięki szyfrowaniu realizowanemu w trybie CBC algorytmu DES.
- Formatowanie wiadomości - definiuje pole `msgSecurityParameters`, obsługujące systemowe funkcje bezpieczeństwa.
- Rozpoznawanie - określa procedury, dzięki którym jednostki SNMP mogą wykrywać wzajemnie swoją obecność w sieci.
- Zarządzanie kluczami - normalizuje techniki generowania, aktualizacji oraz używania kluczy.

1.1.3.6 Polityka sterowania dostępem

Wykorzystanie VACM umożliwia elastyczne konfigurowanie automatu SNMP, tak aby możliwe było

realizowanie właściwego zestawu uprawnień dostępowych, które są określane w odniesieniu do następujących czynników:

- Elementu *principal* żądającego dostępu. VACM umożliwia agentom różnicowanie przywilejów dostępowych poszczególnych użytkowników. Zgodnie z wcześniejszym opisem, elementy *principal* są zazwyczaj łączone w grupy, stąd polityka dostępowa jest zwykle definiowana w odniesieniu do poszczególnych grup.
- Poziomu bezpieczeństwa, na którym żądanie zostaje przekazane w ramach wiadomości SNMP. Najczęściej agent wymaga uwierzytelnienia oraz zaszyfrowania wiadomości z żądaniem realizacji operacji zapisu.
- Modelu bezpieczeństwa użytego w trakcie przetwarzania żądania. Jeśli agent posiada zaimplementowane różnorodne modele bezpieczeństwa, to może być konfigurowany w celu uzyskania innego poziomu dostępu dla każdego z modeli. I tak np. dowolny element może być udostępniany w odpowiedzi na wiadomości realizujące specyfikację USM i nie być dostępny dla żądań realizujących model bezpieczeństwa SNMPv1.
- Wskazany w żądaniu kontekst MIB.
- Specyfiki reprezentacji obiektu, którego żądanie dotyczy (z uwagi na fakt, że znaczenie informacji dotyczących poszczególnych obiektów dla funkcjonowania systemu jako całości jest zróżnicowane).
- Typu żądanego dostępu (czytanie, zapis, powiadomienie).

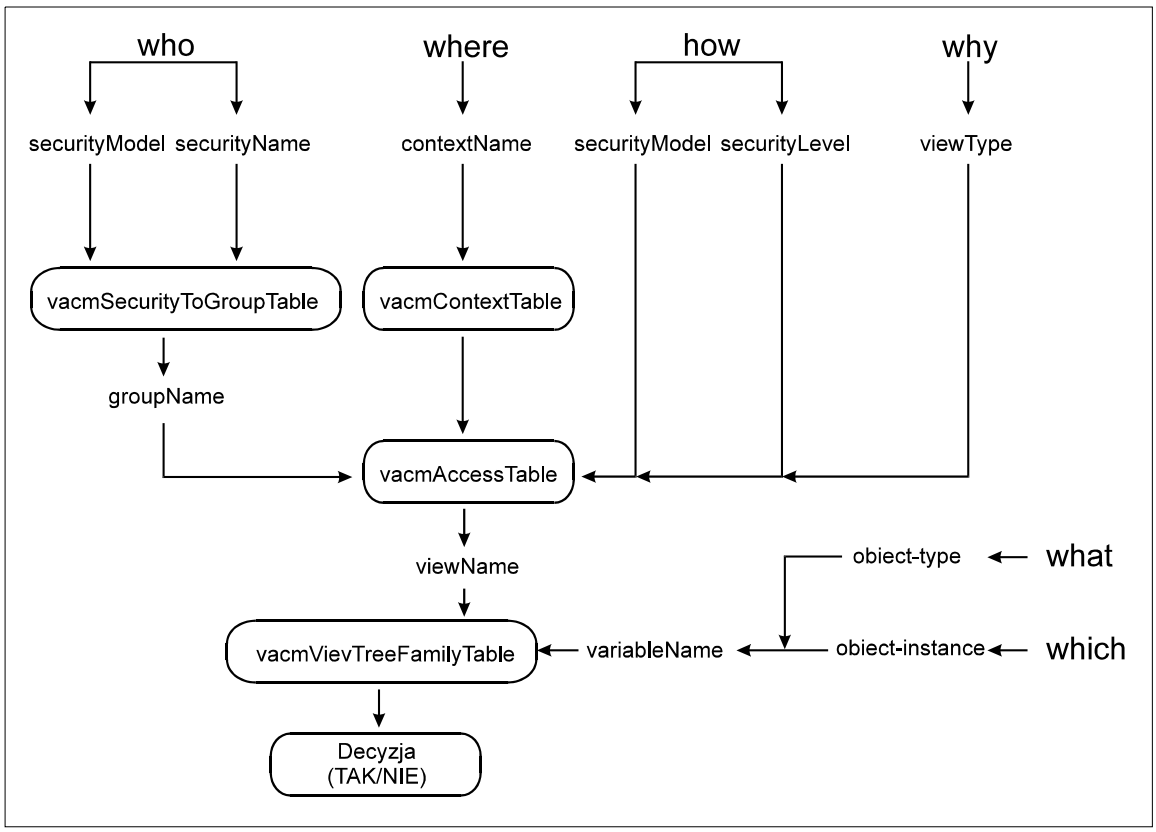
Usługę realizowaną przez Podsystem Sterowania Dostępem określa prymityw `isAccessAllowed` z parametrami:

- `securityModel` - wykorzystywany model bezpieczeństwa;
- `securityName` -żądający dostępu element klasy *principal*;
- `securityLevel` - poziom bezpieczeństwa;
- `viewType` - wizja klasy odczyt, zapis lub powiadomienie;
- `contextName` - kontekst zawierający `variableName`;
- `variableName` - identyfikator OID zarządzanego obiektu.

Parametry te przekazują wszelkie informacje niezbędne dla podjęcia decyzji dostępowej. Jej wynik zwraca prymityw `statusInformation`, który może zawierać jedno z następujących wskazań:

- `accessAllowed` - określona wizja MIB elementu klasy *principal* identyfikowanego przez żądaną `securityName` i inne parametry została odnaleziona, co pozwala na realizację dostępu.
- `notInView` - dana wizja MIB została odnaleziona ale dostęp nie jest możliwy, ponieważ nie zawiera wskazanej wartości `variableName`.
- `noSuchView` - brak wizji wskazanej zawartością `viewType`.
- `noSuchContext` - wskazana wartość `contexName` nie jest obsługiwana.
- `noGroupName` - grupa definiowana przez `securityModel` i `securityName` nie jest obsługiwana.
- `noAccessEntry` - brak wizji MIB odpowiadającej kombinacji `security-Model`, `securityName`, `securityLevel` oraz `contextName`.

Przedstawiony na rys. C.17 schemat ilustruje sposób wykorzystania parametrów wejściowych oraz tryb w jaki proces decyzyjny sterowania dostępem korzysta z tablic bazy danych MIB VACM.



Rys. C.17. Realizacja procesu decyzyjnego

Proces decyzyjny bazuje na następujących informacjach wejściowych:

- **who:** stanowi kombinację pól `securityModel` i `securityName`, określając element *principal*, którego komunikacja jest chroniona przez wskazany model bezpieczeństwa (`securityModel`). Każda z istniejących w ramach automatu SNMP kombinacja jest przypisywana co najmniej jednej grupie przy użyciu obiektu `vacmSecurityToGroupTable`, który odwzorowuje wartość `groupName` na zestaw `securityModel` i `securityName`.
- **where:** określa lokalizację wymaganego obiektu zarządzania wykorzystując obiekt `contextName`, o wartości z zestawu przechowywanego w `vacmContext-Table`.
- **how:** sposób ochrony odebranej PDU typu `Request` lub `Inform` określa kombinacja `securityModel` i `securityLevel`. Zestaw *who, where* i *how* identyfikuje jedno wejście w tablicy `vacmAccessTable` (albo nie wskazuje żadnego).
- **why:** cel uzyskania żądanego dostępu (czytanie, zapis, powiadomienie) określa pole `viewType`. Wybrane wejście `vacmAccessTable` posiada oddzielne nazwy MIB (`viewName`) dla każdego typu operacji, stąd dla wyboru `viewName` niezbędna jest znajomość `viewType`. Wartość `viewName` wskazuje odpowiednią wizję MIB w tablicy `vacmViewTreeFamilyTable`.
- **what:** typ obiektu oraz jego reprezentację wskazują odpowiednio prefiks i sufiks identyfikatora obiektu (`variableName`). Typ obiektu określa przy tym rodzaj (typ) żądanej informacji zarządzania.
- **which:** rodzaj wymaganej informacji jest określany przez reprezentację obiektu.

Ostatecznie, dostęp zostaje przyznany, jeśli w danej wizji MIB występuje element, którego nazwa odpowiada wartości pola `variableName`.

1.1.4 Podsumowanie

SNMPv2 stanowił istotne rozszerzenie SNMPv1, pozostawiające równocześnie pierwotną łatwość rozumienia mechanizmów funkcjonalnych oraz łatwość ich implementacji. Wersja 2 została w szczególności lepiej przystosowania do funkcjonowania w rozproszonym środowisku sieciowym oraz oferuje lepsze parametry użytkowe.

Wspólne niedostatki v1 oraz v2, a zwłaszcza brak profesjonalnie realizowanych funkcji bezpieczeństwa zostały usunięte dzięki zdefiniowaniu wersji 3 SNMP, oferującej poufność, uwierzytelnianie i sterowanie dostępem. Równocześnie, dostawcy oprogramowania uzyskali większe możliwości adaptacyjne, pozwalające na lepsze dostosowanie swoich produktów do wymagań klienta. Mająca nastąpić wkrótce standaryzacja dodatkowych baz danych MIB, która pozwoli na kooperację różnych aplikacji zarządzania sieciowego powinna ugruntować pozycję SNMP na współczesnym rynku telekomunikacyjnym.