Projekt z sieci
Temat: Zarządzanie siecią

Wykonali : Roman Popek
Marcin Weron
Piotr Wolak

1. SNMP – protokół zarządzania sięcią.

W skład system zarządzania sieciowego wchodzą następujące elementy:

- Programy zarządzania
- Agenci
- Bazy informacji i zarządzania(MIB)
- Procedury pośredniczące w komunikacji międzysieciowej

1.1Programy zarządzania **-** oprogramowanie wykorzystywanie do zarządzania sieciowego dające umiejętność asystowania programowi zarządzania czy administratorowi w codziennych czynnościach polegających na kierowaniu siecią lub indywidualnymi urządzeniami. Możliwości i funkcje programu zarządzania sieciowego mogą być pogrupowane w trzy obszary:

- Architektura
  - otwartą i rozszerzoną strukturę
  - wsparcie dla rozproszonego / scentralizowanego monitoringu
  - wsparcie dla wspólnych platform
- podstawowe usługi
  - rozróżnianie i wykrywanie błędów
  - powiadamianie alarmowe i przetwarzanie
  - wspieranie wielu urządzeń
  - narzędzia sprawozdawcze
  - łatwy w użytkowaniu interfejs
  - zarządzanie konfiguracją
  - odkrywanie sieci
- dodatkowe aplikacje
  - etykietowanie problemów
  - zarządzanie oparte na strategii
  - zaawansowane przetwarzanie alarmów
  - symulacja sieci [3]

1.2Agenci - agenci zarządzania sieciowego są modułami oprogramowania rezydujące w urządzeniach sieciowych , takich jak:

- stacje robocze
- drukarki sieciowe
- inne urządzenia sieciowe

Agenci odpowiadają za połączenie między oprogramowaniem zarządzania, a urządzeniem w który agent rezyduje. Kontrolują również funkcjonowanie urządzeń sieciowych przez manipulowanie bazą danych przechowywaną w bazie informacji zarządzania(MIB) znajdującej się wewnątrz urządzenia. Jest również odpowiedzialny za konwertowanie zadań programu zarządzania sieciowego ze standardowego formatu sieciowego, jak i odzyskiwanie pożądanych informacji i odsyłanie wiążących odpowiedzi.[3]

1.3 Bazy informacji i zarządzania(MIB) – zadaniem MIB jest definiowanie bazy danych obiektów, które system zarządzania sieciowego może odzyskać lub którymi może manipulować. Obiekty są dostępne dla systemu zarządzania sieciowego poprzez agenta. [3]

1.4 Procedury pośredniczące w komunikacji międzysieciowej (Proxy) – wykorzystywane są do wypełnienia luki pomiędzy standardowymi programami zarządzania protokołami sieciowymi a systemami nie zapewniających takich

standardów. Systemy oparte na proxy mogą zapewniać łączność z systemami które nie posiadają standardowych protokołów sieciowych. [3]

2. Protokół SNMP obejmuje obszary funkcjonalne zarządzania:

2.1 Zarządzanie konfiguracją - identyfikuje, sprawdza i dostarcza dane w celu odpowiedniego przygotowania do inicjowania, startowania i zapewnienia ciągłości funkcjonowania obiektu. Obejmuje m.in. ustawianie wartości atrybutów, które sterują funkcjonowaniem obiektu, zbieranie informacji o bieżącym jego stanie oraz zmieniania konfiguracji obiektu. [3]
2.2 Zarządzanie obsługą uszkodzeń - wykrywanie niesprawności, wyizolowanie i korekta niepoprawnego działania obiektu. Uszkodzenia powodują, że obiekt nie realizuje swoich zadań, przy czym mogą być one trwałe lub przejściowe. Funkcje zarządzające w tym zakresie obejmują m.in.: potwierdzanie i podejmowanie akcji po zgłoszeniu wykrycia błędu, śledzenie i identyfikacja niesprawności. [3]
2.3 Zarządzanie efektywnością - umożliwia oszacowanie zachowania się obiektu i skuteczności jego funkcjonowania. Obejmuje m.in. funkcje zbierania informacji statystycznych, określenie efektywności obiektu w warunkach naturalnych, sztucznych, zmianę trybu pracy obiektu w celu przeprowadzenia działań związanych z zarządzaniem efektywnością. [3]
2.4. Zarządzanie bezpieczeństwem - obejmuje funkcje tworzenia, usuwania i sterowania mechanizmami zabezpieczeń, raportowania zdarzeń związanych z zabezpieczeniem oraz rozdzielania informacji związanych z zabezpieczeniem. [3]
2.5 Zarządzanie rozliczeniami - umożliwia ustalanie taryfikacji za dostęp do obiektu i korzystanie z jego zasobów. Funkcja zarządzania rozliczeniami dotyczy m.in. informowania o kosztach wynikających z wykorzystania obiektu. [3]

Oprócz protokołów, które zapewniają usługi na poziomie sieci i programów użytkowych umożliwiających wykorzystywanie tych protokołów oprogramowanie intersieci potrzebuje narzędzi pozwalających administratorom na znajdowanie przyczyn problemów, sterowanie wyznaczaniem tras oraz wykrywanie komputerów, które naruszają standardy protokołów. Zadania tego typu określa się mianem zarządzania intersiecią. Obecnie standardowym protokołem zarządzania sieciami TCP/IP jest SNMP (*Simple Network Management Protocol)*. [3]

2.6 **Standard dla informacji sterujących.**
Router kontrolowany za pomocą SNMP musi przechowywać i udostępniać administratorowi informacje sterujące i informacje o stanie. Router taki gromadzi np. informacje statystyczne dotyczące stanu jego interfejsów sieciowych, przychodzących i wysyłanych pakietów, porzucanych datagramów oraz wysyłanych komunikatów o błędach. Protokół SNMP umożliwia administratorowi dostęp do tych informacji, ale nie określa, w jaki sposób ten dostęp ma być zrealizowany. Temu zagadnieniu jest poświęcony osobny standard -MIB (*Management Information Base*), który określa jakie informacje musi przechowywać router i jakie operacje mogą być na nich określone. Standard MIB np. wymaga, aby oprogramowanie IP przechowywało informacje o liczbie oktetów przychodzących do każdego z interfejsów sieciowych, określa też że oprogramowanie zarządzające może jedynie odczytywać tę informację.
Standard MIB dla TCP/IP dzieli informacje sterujące na osiem kategorii, wymienionych w tabelce:

Lp	Kategoria MIB	Związane informacje
1	System	System operacyjny komputera lub routera
2	Interfaces	Poszczególne interfejsy sieciowe
3	addr.trans.	Tłumaczenie adresów (np. odwzorowanie ARP)
4	Ip	Oprogramowanie IP
5	Icmp	Oprogramowanie ICMP
6	Tcp	Oprogramowanie TCP
7	Udp	Oprogramowanie UDP
8	Egp	Oprogramowanie EGP

Wybór tych kategorii jest istotny, gdyż identyfikator każdej pozycji MIB zawiera kod tej kategorii.
Oddzielenie standardu MIB od protokołu zarządzania siecią ma zalety zarówno dla użytkowników, jak i dla producentów sprzętu i oprogramowania. Producent może dostarczać oprogramowanie SNMP wraz z routerem i mieć pewność, że oprogramowanie to będzie działać poprawnie nawet po określeniu nowych pozycji MIB. Użytkownik może używać tego samego klienta zarządzania siecią do sterowania pracą wielu routerów wyposażonych w różne wersje MIB. Oczywiście router, który nie ma nowych pozycji MIB nie może dostarczyć żądanych informacji, mogą więc analizować używają tego samego języka do przekazywania informacji, mogą więc analizować zapytania i dostarczać żądanych informacji lub wysyłać komunikat, że nie mają potrzebnych pozycji. [1]

3. Protokół SNMP.

Protokoły zarządzania siecią określają sposób komunikacji między programem-klientem do zarządzania siecią używanym przez administratora a serwerem działającym na komputerze bądź routerze. Protokoły zarządzania siecią definiują również zależności między kontrolowanymi routerami. Oznacza to, że udostępniają one możliwość uwierzytelniania administratorów.
SNMP jest dość stabilny, gdyż jego definicja nie wymaga zmian przy dodawaniu nowych zmiennych do MIB i określaniu operacji na nich (gdyż są one definiowane jako skutki uboczne zmiany wartości tych zmiennych). Protokół SNMP jest łatwy do zrozumienia i implementacji. Umożliwia łatwe wykrywanie błędów, gdyż nie wymaga osobnych przypadków szczególnych dla poszczególnych poleceń. SNMP jest bardzo elastyczny, gdyż umożliwia dodawania nowych poleceń w ramach bardzo eleganckiego mechanizmu.

SNMP zapewnia więcej niż tylko dwie operacje:

Polecenie	Znaczenie
GetRequest	Odczytaj wartość wskazanej zmiennej (otrzymaj żądanie)
GetNextRequest	Odczytaj wartość bez podania jej dokładnej nazwy (otrzymaj następne żądanie)
GetResponse	Odpowiedź na operację odczytania wartości (otrzymaj odpowiedz)
SetRequest	Ustaw wartość określonej zmiennej (ustaw żądanie)
Trap	Odpowiedź wywołana przez zdarzenie (wychwytuj zdarzenia)

Operacje *GetRequest, GetResponse* i *SetRequest* stanowią podstawowy zestaw operacji do odczytywania i zapisywania danych (oraz odpowiedzi na te operacje). SNMP

wymaga, by operacje były niepodzielne (*atomic*), co znaczy, że jeśli komunikat SNMP nakazuje wykonanie operacji dotyczących wielu zmiennych, to serwer musi wykonać wszystkie nakazane operacje, albo żadnej. W szczególności w razie wystąpienia błędu nie zmieni się wartość żadnej ze zmiennych. Operacja *Trap* umożliwia administratorowi wyrażanie żądania, by serwer wysyłał informacje, gdy wystąpią określone zdarzenia. Można np. nakazać serwerowi SNMP wysyłanie do administratora komunikatu *Trap*, gdy jedna z sieci stanie się niedostępna (np. z powodu awarii interfejsu).
Protokół SMNP jest rozszerzalny. Dostawcy mogą poszerzać zakres informacji gromadzonych w bazie MIB, tak aby możliwe było uwzględnienie cech urządzeń przez nich produkowanych.[3]

3.1 Format komunikatów SNMP.

W odróżnieniu od większości protokołów TCP/IP komunikaty SNMP nie mają ustalonych pól, lecz używają standardowego kodowania ASN.1. Komunikat SNMP składa się z trzech głównych części: *version* (wersja protokołu), *community* (identyfikator wspólnoty SNMP służący do grupowania routerów zarządzanych przez danego administratora) oraz *data* (obszaru danych). Obszar danych jest podzielony na jednostki danych protokołu (*Protocol Data Unit, PDN*). Każda jednostka składa się z żądania (wysyłanego przez klienta) lub odpowiedzi (wysyłanej przez serwer). [1]

3.2 Kody odpowiedzi SNMP

Kody błędów przesyłane z powrotem od agenta SNMPv1 były bardzo ograniczone. Na przykład, program zarządzania chciał wykonać obiekty MIB, a agent nie mógł wykonać tego polecenia, odpowiedział to komunikatem *noSuchName* (nie ma takiej nazwy). Dzięki dodatkowym kodom błędów w SNMPv2 agent w takiej sytuacji odpowiedziałby komunikatem *notWritable* . Lista kodów odpowiedzi przedstawiona jest w tabeli poniżej: [1]

Kody odpowiedzi	**Znaczenie**
SNMPv1	
TooBig	Przesyłany z powrotem przez agenta jeśli odpowiedz na żądanie ma za dużą objętość, aby ją przesłać.
NoSuchName	Przesyłany z powrotem przez agenta w jednym z dwóch przypadków: 1) jeśli operacji ustawiania dokonuje się na obiekcie, który nie jest w widoku bazy MIB; 2) jeśli operacji ustawiania dokonuje się na obiekcie, który znajduje się w widoku bazy MIB, ale obiekt jest tylko do odczytu. Ten kod jest teraz wykorzystywany dla zgodności proxy.
BadValue	Przesyłany z powrotem przez agenta, który odkrył błąd w liście powiązań zmiennych PDU. Ten kod jest teraz wykorzystywany dla zgodności proxy.
re*ad-only*	Przesyłany z powrotem przez agenta . Ten kod jest teraz wykorzystywany dla zgodności proxy.
GenError	Przesyłany z powrotem przez agenta, kiedy nie powiedzie się przetwarzanie PDU z innego powodu niż te które są wymienione w tabeli
SNMPv2	
NoAccess	Zmienna znajduje się poza zdefiniowanym zasięgiem bazy MIB, co uniemożliwia wykonanie czynności.
NotWritable	Zmienna istnieje wewnątrz agenta, ale agent nie jest w stanie

	zmodyfikować obiektu.
WrongType	Dostarczana wartość jest złym typem danych tak, jak jest zdefiniowane przez ASN.1.
WrongLenght	Dostarczana wartość ma nieodpowiednią długość.
WrongEncoding	Dostarczana wartość nie była prawidłowo zakodowana.
WrongValue	Dostarczana wartość nie znajduje się w zakresie wymaganym dla tego typu obiektu.
Nocreation	Obiekt nie istnieje i agent nie jest w stanie stworzyć przykładu tego obiektu.
InconsistentName	Obiekt nie istnieje i agent nie może stworzyć przykładu tego obiektu, ponieważ nazwa jest nie zgodna z wartościami powiązanych obiektów.
InconsistentValue	Dostarczany obiekt jest niezgodny z wartościami zarządzanych obiektów.
resourceUnavailable	Potrzebne źródło nie może zostać zachowane, aby zaspokoić żądanie.

3.3 Transmisja wiadomości SNMP

Kiedy program zarządzania siecią definiuje wiadomość SNMP, mają miejsce następujące wydarzenia:

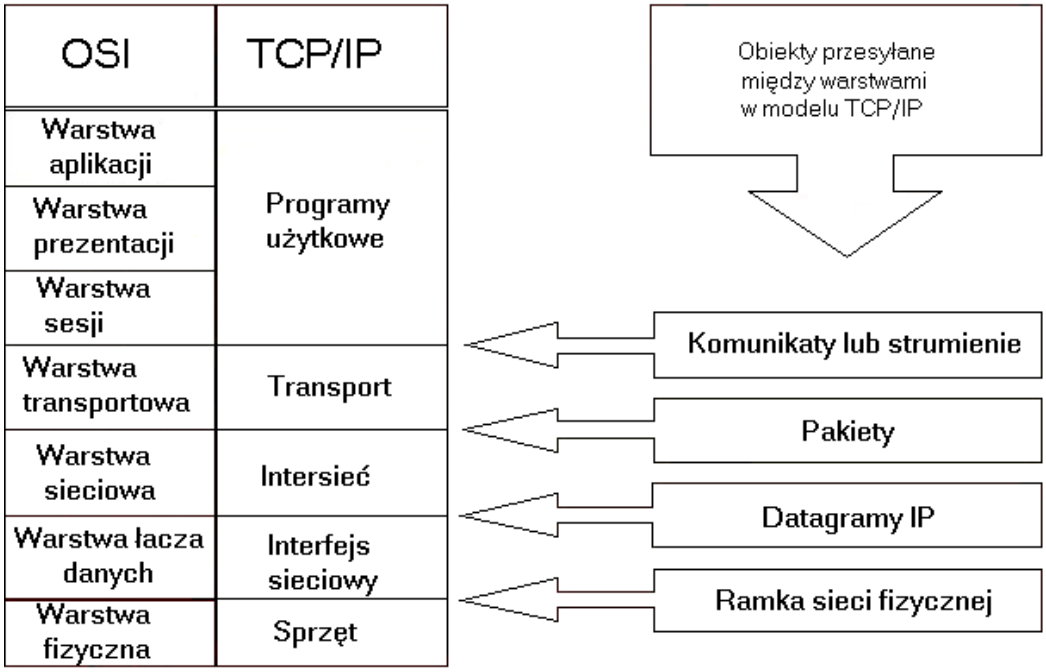
- Podstawowa jednostka PDU jest skonstruowana.
- PDU jest przesyłana do warstwy obsługi bezpieczeństwa , jeśli jest ona dostępna.
- Warstwa protokołu formatuje wiadomość włącznie z informacją o wersji i społeczności.
- Cała wiadomość jest teraz zakodowana przy użyciu zasad ASN.1.
- Wiadomość jest przesyłana do usług transportowych tak, aby mogła być dostarczona do pozostałych elementów.

Kiedy urządzenie agenta otrzymuje wiadomość SNMP, mają miejsce poniższe wydarzenia:

- Podstawowa kontrola jest przeprowadzana w celu upewnienia się, że wiadomość jest sformatowana poprawnie. Wiadomość jest odrzucana, jeśli pojawią się jakieś błędy.
- Informacja o wersji protokołu jest weryfikowana. Jeśli istnieje jakaś niezgodność, wiadomość jest odrzucana.
- Serwis bezpieczeństwa próbuje weryfikować wysyłające elementy. Jeśli mu się to nie uda, uruchamiane jest wychwytywanie zdarzeń, a wiadomość jest odrzucana.
- Jednostka PDU jest rozkodowywana.
- Jednostka PDU jest przetwarzana. [3]

4. Pakiet protokołu TCP/IP

4.1 TCP/IP a model OSI



TCP/IP (Transmission Control Protocol/Internet Protocol) nie jest pojedynczym produktem. Jest to uogólniona nazwa całej rodziny protokołów i oprogramowania udostępniającego szereg usług sieciowych. Może zostać wykorzystany w dowolnym zbiorze połączonych ze sobą sieci. Technika TCP/IP stanowi podstawowe rozwiązanie światowej intersieci - INTERNETU. Protokół TCP/IP ma strukturę warstwową i ma do niego zastosowanie większość filozofii modelu OSI. Warstwy TCP/IP różnią się jednak od warstw OSI, o czym się zaraz przekonamy. Protokoły TCP i IP ustalają zasady komunikacji - opisują szczegóły formatu komunikatów, sposób odpowiadania na otrzymany komunikat, określają jak komputer ma obsługiwać pojawiające się błędy lub inne nienormalne sytuacje.
Umożliwiają one rozpatrywanie zagadnień dotyczących komunikacji niezależnie od sprzętu sieciowego.
Zapewniają szereg popularnych usług dostępnych dla użytkowników:

- poczta elektroniczna
- przesyłanie plików
- praca zdalna [3]

4.2 Zadania warstw w TCP/IP

- Warstwa programów użytkowych - na najwyższym poziomie użytkownicy wywołują programy użytkowe, które mają dostęp do usług TCP/IP. Programy użytkowe współpracują z jednym z protokołów na poziomie warstwy transportowej i wysyłają lub odbierają dane w postaci pojedynczych komunikatów lub strumienia bajtów.
- Warstwa transportowa - jej podstawowym zadaniem jest zapewnienie komunikacji między jednym programem użytkownika a drugim. Warstwa ta może regulować przepływ informacji. Może też zapewnić pewność przesyłania. W tym celu organizuje wysyłanie przez odbiorcę potwierdzenia otrzymania oraz ponowne wysyłanie utraconych pakietów przez nadawcę.
- Warstwa intersieci - odpowiada za obsługę komunikacji jednej maszyny z drugą. Przyjmuje ona pakiety z warstwy transportowej razem z informacjami

identyfikującymi maszynę - odbiorcę, kapsułkuje pakiet w datagramie IP, wypełnia jego nagłówek, sprawdza czy wysłać datagram wprost do odbiorcy czy też do routera i przekazuje datagram do interfejsu sieciowego. Warstwa ta zajmuje się także datagramami przychodzącymi, sprawdzając ich poprawność i stwierdzając czy należy je przesłać dalej czy też przetwarzać na miejscu.

- Warstwa interfejsu sieciowego - odbiera datagramy IP i przesyła je przez daną sieć. [3]

4.3 Własności usługi niezawodnego dostarczania

TCP organizuje dwukierunkową współpracę między warstwą IP, a warstwami wyższymi, uwzględniając przy tym wszystkie aspekty priorytetów i bezpieczeństwa. Musi prawidłowo obsłużyć niespodziewane zakończenie aplikacji, do której właśnie wędruje datagram, musi również bezpiecznie izolować warstwy wyższe - w szczególności aplikacje użytkownika - od skutków awarii w warstwie protokołu IP. Scentralizowanie wszystkich tych aspektów w jednej warstwie umożliwia znaczną oszczędność nakładów na projektowanie oprogramowania.
TCP rezyduje w modelu warstwowym powyżej warstwy IP. Warstwa ta jest jednak obecna tylko w tych węzłach sieci, w których odbywa się rzeczywiste przetwarzanie datagramów przez aplikacje, tak więc nie posiadają warstwy TCP na przykład routery, gdyż warstwy powyżej IP nie miałyby tam nic do roboty. [3]

4.5 Realizacja niezawodnego połączenia

Aby zagwarantować, że dane przesyłane z jednej maszyny do drugiej nie są tracone, ani duplikowane używa się podstawowej metody znanej jako pozytywne potwierdzanie z retransmisją. Metoda ta wymaga, aby odbiorca komunikował się z nadawcą, wysyłając mu w momencie otrzymania danych komunikat potwierdzenia (ACK). Nadawca zapisuje sobie informację o każdym wysłanym pakiecie i przed wysłaniem następnego czeka na potwierdzenie. Oprócz tego nadawca uruchamia zegar w momencie wysyłania pakietu i wysyła ten pakiet ponownie, gdy minie odpowiedni czas, a potwierdzenie nie nadejdzie. [3]

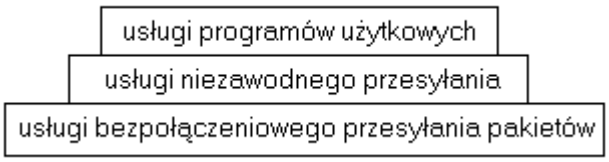
4.6 Kontrola przepływu danych

TCP zapewnia wydajną transmisję danych poprzez sieć dzięki kontroli przepływu danych. Odkrywa ona dynamicznie charakterystykę opóźnień sieci i reguluje jej działanie, aby maksymalizować przepustowość bez przeciążania sieci.
Każdy węzeł końcowy połączenia TCP ma bufor, służący do zapamiętywania danych transmitowanych poprzez sieć, dopóki dana aplikacja nie będzie gotowa do odczytu tych danych. Pozwala to, aby mogły mieć miejsce przesyły sieciowe w czasie, kiedy aplikacje są zajęte innymi procesami i poprawia ogólną wydajność. TCP zarządza ruchem tak, aby jego bufory się nie przepełniały — szybcy nadawcy są okresowo zatrzymywani, żeby wolniejsi nadawcy mogli nadążyć.
Aby uniknąć przepełnienia bufora, TCP ustawia pole rozmiar okna w każdym z pakietów, które transmituje. Pole to wskazuje ilość danych, które mogą być transmitowane do bufora. Gdy wartość ta spadnie do zera, to host transmitujący nie będzie przesyłał żadnych danych, dopóki nie otrzyma pakietu anonsującego wartość niezerową w polu rozmiaru okna.
Czasami przestrzeń bufora jest zbyt mała do wydajnej transmisji. Zdarza się to w sieciach, które mają ograniczoną przepustowość albo powolne łącza. Rozwiązaniem jest zwiększenie rozmiaru bufora, lecz istnieje w tej kwestii ograniczenie narzucone przez

maksymalny rozmiar okna, jaki dopuszcza protokół. W takiej sytuacji określa się sieć jako sieć LFN (*Long Fat Network*).
Jednym z ważnych czynników rządzących przepływem informacji poprzez sieć jest okres czasu, przez jaki host wysyłający czeka na potwierdzenie, zanim założy, że dane uległy zagubieniu i dokona ponownej ich transmisji. Jeżeli okres ten jest zbyt krótki, to pakiety są niepotrzebnie retransmitowane; jeżeli jest on zbyt długi, to dane połączenie będzie stało bezczynnie, podczas gdy host będzie czekał, dopóki okres nie minie. TCP podejmuje próbę ustalenia optymalnego okresu wygaśnięcia poprzez monitorowanie normalnej wymiany pakietów danych. Proces ten zwany jest szacowaniem czasu przewidzianego na transmisję i potwierdzenie przyjęcia (RTT). [3]

4.7 IP (Internet Protocol)

Zasadniczo sieć TCP/IP udostępnia trzy zbiory usług (rys poniżej) [3]:



Najbardziej podstawowa usługa - przenoszenie pakietów bez użycia połączenia nosi nazwę Internet Protocol, a zwykle oznacza się skrótem IP. Usługa ta jest zdefiniowana jako zawodny (ang. unreliable) system przenoszenia pakietów bez użycia połączenia, tzn. nie ma gwarancji, że przenoszenie zakończy się sukcesem. Każdy pakiet obsługiwany jest niezależnie od innych.
Pakiety z jednego ciągu, wysyłanego z danego komputera do drugiego, mogą podróżować różnymi ścieżkami, niektóre z nich mogą zostać zgubione, inne natomiast dotrą bez problemów.
Pakiet może zostać zagubiony, zduplikowany, zatrzymany, lub dostarczony z błędem, a system nie sprawdzi, że coś takiego zaszło, a także nie powiadomi o tym ani nadawcy, ani odbiorcy.

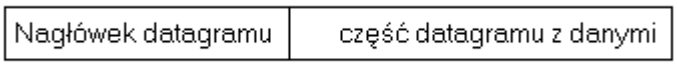
Protokół IP zawiera trzy definicje:

- Definicję podstawowej jednostki przesyłanych danych, używanej w sieciach TCP/IP. Określa ona dokładny format wszystkich danych przesyłanych przez sieć.
- Definicję operacji trasowania, wykonywanej przez oprogramowanie IP, polegającej na wybieraniu trasy, którą będą przesyłane dane.
- Zawiera zbiór reguł, które służą do realizacji zawodnego przenoszenia pakietów. Reguły te opisują, w jaki sposób węzły i routery powinny przetwarzać pakiety, jak i kiedy powinny być generowane komunikaty o błędach oraz kiedy pakiety mogą być porzucane. [3]

4.8 Datagram IP

Podstawowa jednostka przesyłanych danych nazywana jest datagramem. Datagram podzielony jest na nagłówek i dane (rysunek poniżej). Nagłówek datagramu zawiera adres nadawcy i odbiorcy oraz pole typu, które identyfikuje zawartość datagramu. Datagram przypomina ramkę sieci fizycznej. Różnica polega na tym, że nagłówek ramki zawiera adresy fizyczne, zaś nagłówek datagramu adresy IP (omówione później).

Ponieważ przetwarzaniem datagramów zajmują się programy, zawartość i format datagramów nie są uwarunkowane sprzętowo. [3]



4.9 Adresy IP

Adres IP jest 32-bitową liczbą całkowitą zawierającą informacje o tym do jakiej sieci włączony jest dany komputer, oraz jednoznaczny adres w tej sieci. Zapisywany jest on w postaci czterech liczb dziesiętnych oddzielonych kropkami, przy czym każda liczba dziesiętna odpowiada 8 bitom adresu IP. np. 32-bitowy adres 10000000 00001010 00000010 00011110 jest zapisany jako 128.10.2.30 . Adresy IP podzielone są na klasy. Klasa adresu IP określona jest przez najstarsze bity, przy czym do zidentyfikowania jednej z trzech zasadniczych klas (A, B, C) wystarczą dwa pierwsze bity. Taki mechanizm adresowania wykorzystują routery, które używają adresu sieci do wyznaczania trasy pakietów.

Klasy adresów IP [2]

Obserwując najstarsze bity adresu można stwierdzić do jakiej klasy należy dany adres, w efekcie można stwierdzić ile bitów będzie adresowało sieć, ile zaś sam komputer.
Łatwo zauważyć, że adresów klasy A jest niewiele (2^7=128), ale w każdej z sieci tej klasy może być aż 65535 maszyn.
Klasa B to 2^{14} sieci i 2^{16} komputerów.
W klasie C sieć adresowana jest za pomocą 21 bitów - daje to 2^{21} sieci, ale w każdej z nich może być co najwyżej 2^8=256 maszyn.
Adres klasy D ma specjalne znaczenie - jest używany w sytuacji gdy ma miejsce jednoczesna transmisja do większej liczby urządzeń.
Jak wspomniano, adresy zamiast w postaci bitowej, zwykle zapisuje się w postaci czterech liczb dziesiętnych. Wówczas podział na klasy wygląda następująco:

klasa	**Najniższy adres**	**Najwyższy adres**

A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Przydzielanie adresów sieciowych
W celu zapewnienia jednoznaczności identyfikatorów sieci, wszystkie adresy przydzielane są przez jedną organizację. Zajmuje się tym INTERNIC (Internet Network Information Center). Przydziela ona adresy sieci, zaś adresy maszyn administrator może przydzielać bez potrzeby kontaktowania się z organizacją. Organizacja ta przydziela adresy tym instytucjom, które są lub będą przyłączone do ogólnoświatowej sieci INTERNET. Każda instytucja może sama wziąć odpowiedzialność za ustalenie adresu IP, jeśli nie jest połączona ze światem zewnętrznym. Nie jest to jednak dobre rozwiązanie, gdyż w przyszłości może uniemożliwić współpracę między sieciami i sprawiać trudności przy wymianie oprogramowania z innymi ośrodkami.

5 Protokół UDP

W zestawie protokołów TCP/IP protokół datagramów użytkownika UDP (ang. User Datagram Protocol), zapewnia porty protokołów używane do rozróżniania programów wykonywanych na pojedynczej maszynie. Oprócz wysyłanych danych, każdy komunikat zawiera numer portu odbiorcy i numer portu nadawcy, dzięki czemu oprogramowanie UDP odbiorcy może dostarczyć komunikat do właściwego adresata. Do przesyłania komunikatów między maszynami UDP używa podstawowego protokołu IP i ma tę samą niepewną, bezpołączeniową semantykę dostarczania datagramów co IP - nie używa potwierdzeń w celu upewnienia się, o dotarciu komunikatów i nie zapewnia kontroli szybkości przesyłania danych między maszynami. Z tego powodu komunikaty UDP mogą być gubione, duplikowane lub przychodzić w innej kolejności niż były wysłane, ponadto pakiety mogą przychodzić szybciej niż odbiorca może je przetworzyć. Program użytkowy korzystający z UDP musi na siebie wziąć odpowiedzialność za rozwiązanie problemów niezawodności.
Ponieważ sieci lokalne dają dużą niezawodność i małe opóźnienia wiele programów opartych na UDP dobrze pracuje w sieciach lokalnych, ale może zawodzić w większych intersieciach TCP/IP. [3]

6 Protokół ARP i RARP

Protokół odwzorowania adresów (ARP)

Opisaliśmy już schemat adresowania TCP/IP, w którym każdy komputer ma przypisany 32-bitowy adres jednoznacznie identyfikujący go w sieci. Jednak dwie maszyny mogą się komunikować tylko wtedy kiedy znają nawzajem swoje adresy fizyczne. Zachodzi więc potrzeba przekształcenia adresu IP na adres fizyczny tak aby informacja mogła być poprawnie przesyłana.Problem ten przedstawimy na przykładzie sieci ethernet, w której mamy do czynienia z długim 48-bitowym adresem fizycznym przypisanym w trakcie procesu produkcyjnego urządzeń sieciowych. W efekcie podczas wymiany karty sieciowej w komputerze, zmienia się adres fizyczny maszyny. Ponadto nie ma sposobu na zakodowanie 48-bitowego adresu ethernetowego w 32-bitowym adresie IP.

Przekształcenia adresu IP na adres fizyczny dokonuje protokół odwzorowania adresów ARP (Address Resolution Protocol), który zapewnia dynamiczne odwzorowanie i nie wymaga przechowywania tablicy przekształcania adresowego. [3]

6.2 Protokół odwrotnego odwzorowania adresów (RARP)

Wiemy już jak maszyna może uzyskać adres fizyczny innego komputera, znając jego adres IP. Adres IP jest zwykle przechowywany w pamięci zewnętrznej komputera, skąd jest pobierany w trakcie ładowania systemu operacyjnego. Nasuwa się więc pytanie: jak maszyna nie wyposażona w dysk twardy określa swój adres IP? Odpowiedź: w sposób przypominający uzyskiwanie adresu fizycznego. Protokół odwrotnego odwzorowania adresów RARP (Reverse Address Resolution Protocol) umożliwia uzyskiwanie adresu IP na podstawie znajomości własnego adresu fizycznego (pobranego z interfejsu sieciowego).
Komputery bez dysku twardego pobierają adres IP z maszyny uprawnionej do świadczenia usług RARP, po przesłaniu zapytania z własnym adresem fizycznym. [3]

7 Protokół ICMP

Jak już wiemy oprogramowanie Internet Protocol realizuje zawodne przenoszenie pakietów bez użycia połączenia. Datagram wędruje od nadawcy przez różne sieci i routery aż do końcowego odbiorcy. Jeżeli router nie potrafi ani wyznaczyć trasy ani dostarczyć datagramu, albo gdy wykrywa sytuacje mającą wpływ na możliwość dostarczenia datagramu np. przeciążenie sieci, wyłączenie maszyny docelowej, wyczerpanie się licznika czasu życia datagramu to musi poinformować pierwotnego nadawcę, aby podjął działania w celu uniknięcia skutków tej sytuacji. Protokół komunikatów kontrolnych internetu ICMP (ang. Internet Control Message Protocol) powstał aby umożliwić routerom oznajmianie o błędach oraz udostępnianie informacji o niespodziewanych sytuacjach. Chociaż protokół ICMP powstał, aby umożliwić routerom wysyłanie komunikatów to każda maszyna może wysyłać komunikaty ICMP do dowolnej innej. Protokół ICMP jest traktowany jako wymagana część IP i musi być realizowany przez każdą implementację IP.
Z technicznego punktu widzenia ICMP jest mechanizmem powiadamiania o błędach. Gdy datagram powoduje błąd, ICMP może jedynie powiadomić pierwotnego nadawcę o przyczynie. Nadawca musi otrzymaną informację przekazać danemu programowi użytkownika, albo podjąć inne działanie mające na celu uporanie się z tym problemem. Każdy komunikat ICMP ma własny format, ale wszystkie zaczynają się trzema takimi samymi polami:
8-bitowe pole TYP komunikatu identyfikuje komunikat,
8-bitowe pole KOD daje dalsze informacje na temat rodzaju komunikatu,
Pole SUMA KONTROLNA (obliczane podobnie jak suma IP, ale suma kontrolna ICMP odnosi się tylko do komunikatu ICMP).
Oprócz tego komunikaty ICMP oznajmiające o błędach zawsze zawierają nagłówek i pierwsze 64 bity danych datagramu, z którym były problemy. [3]

8 Narzędzia i programy użytkowe systemu.

Przedstawimy zbiór narzędzi i programy użytkowe z których korzystałby administrator sieci.
Wyróżniamy tu:

- ARP
- IFCONFIG
- NETSTAT
- PING

Krótka charakterystyka poleceń

8.1 NARZĘDZIE ARP

Polecenie `arp` jest używane do wyświetlania i do manipulowania tabelą adresów w lokalnym systemie Unix. Tabela rozróżniania adresów, znana jest również jako pamięć podręczna ARP, zawiera listę wszystkich protokołów łącza danych wykorzystywanych do odwzorowywania adresów IP dla potrzeb sieci lokalnej.
To polecenie zapewnia możliwość przeglądania i modyfikowania pamięci podręcznej ARP. Za pomocą polecenia `arp` można :

- wyświetlić pamięć podręczną ARP
- skasować pozycję ARP
- dodać pozycję ARP

Opcje wiersza poleceń:

- a - wyświetla bieżącą pamięć podręczną ARP
- d - kasuje pozycję ARP
- f -używa do ładowania pliku, który zawiera pozycje przeznaczone do umieszczenia w pamięci podręcznej
- s - tworzy pozycję ARP

Wyświetlanie pamięci podręcznej ARP

Chcąc wyświetlić zawartość tabeli ARP, używamy polecenia `arp -a` . Polecenie przedstawia urządzenie, adres IP, fizyczny adres. Wynik tego polecenia jest przedstawiony poniżej:

```
Debek:# arp -a

alfar (192.168.11.206) at 00:E0:7D:B0:0B:B9 [ether] on eth0
```

Kasowanie pamięci podręcznej

Wykorzystywane jest do usuwania jednej lub więcej pozycji z tabeli ARP, np. jeśli zmieniamy kartę interfejsu sieci na nową to adres sprzętu sieciowego zostaje zmieniony. W tym przypadku, aby wykonać tę operację korzystamy z opcji `arp -d` w celu wykasowania pozycji ARP.

```
Debek:~# arp -d alfar
Debek:~# arp -a
alfar (192.168.11.206) at <incomplete> on eth0

alfar – jest to nazwa hosta
```

Dołączanie pozycji do pamięci podręcznej
Konieczność dopisania pozycji do tabeli jest, gdy konieczna jest komunikacja z danym urządzeniem, a urządzenie nie zapewnia ARP lub jego wdrożenie jest niefunkcjonalne. Aby wykonać dołączenia wykorzystujemy do tego celu polecenie `arp –s .`

```
Debek:~# arp -s alfar 00:E0:7D:B0:0B:b B9
```

Jeśli nie pojawi się żadne polecenie o błędzie, oznaczać to będzie , że polecenie zostało wykonane poprawnie .

8.2 NARZĘDZIE IFCONFIG

Polecenie `ifconfig` jest wykorzystywane do krótkookresowej konfiguracji interfejsu sieci lokalnych. To polecenie pozwala na wykonanie następujących czynności:

- wyświetla listę konfiguracji każdego zdefiniowanego interfejsu sieci.
- Uruchamia / unieruchamia zdefiniowany interfejs sieci.
- Modyfikuje parametry konfiguracji interfejsu sieci.

```
Debek:~# ifconfig

eth0      Link encap:Ethernet  HWaddr 00:20:AF:42:98:04
          inet addr:192.168.11.207  Bcast:192.168.11.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2075 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:209671 (204.7 KiB)  TX bytes:2493 (2.4 KiB)
          Interrupt:10 Base address:0x300
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:3924  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:560 (560.0 b)  TX bytes:560 (560.0 b)
```

Za pomocą `ifconfig` mogą zostać zmienione następujące informacje:

- Adres IP
- Maska sieci
- Adres sieciowy
- Adres łącza danych
- MTU

8.3.NARZĘDZIE NETSTAT

Polecenie `netstat` dostarcza informacji dotyczących obecnego stanu połączenia sieciowego, trasowania i innych ważnych spraw związanych z siecią. Jest ściśle przeznaczone do monitorowania, oraz jest popularnym narzędziem do usuwania błędów.

Podział narzędzia `netstat` na wiele kategorii ze względu na jego funkcjonalność:

- Aktywne sesje sieciowe
- Informacje i dane statystyczne dotyczące interfejsu
- Informacje o tabeli trasowania
- Struktury danych sieciowych
- Rozmaite opcje wyświetlania

- Wyświetlanie aktywnych sesji

Jedną ze znaczących usług zapewnionych przez `netstat` jest zdolność do przeglądania aktywnych połączeń pomiędzy segmentami. Każde aktywne połączenie TCP między lokalnym hostem a innym systemem może być monitorowane. Polecenie to może zostać wykorzystane do szybkiego kontrolowania i upewnienia się, że w danym systemie są prawidłowe usługi. W celu wyświetlenia bieżącego połączenia, wykonujemy połączenie `netstat` bez żadnych parametrów i otrzymujemy:

```
Debek:~# netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 debek.promien.prz.r:ssh alfar:1040          ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  3      [ ]         DGRAM                    170    /dev/log
unix  1      [ W ]       STREAM     CONNECTED     306
unix  1      [ ]         STREAM     CONNECTED     305
unix  0      [ ]         DGRAM                    232
unix  0      [ ]         DGRAM                    204
unix  0      [ ]         DGRAM                    174

Aktywne gniazda
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node Path
unix  3      [ ]         DGRAM                    170    /dev/log
unix  1      [ W ]       STREAM     CONNECTED     306
unix  1      [ ]         STREAM     CONNECTED     305
unix  0      [ ]         DGRAM                    232
unix  0      [ ]         DGRAM                    204
unix  0      [ ]         DGRAM                    174
```

Otrzymaliśmy dwa rodzaje połączeń:

1. TCP – zawiera informacje dotyczące lokalnego i zdalnego adresu, informacje statyczne i stan połączenia.
2. Połączenia obejmujące interfejsy gniazd strumienia.

W celu uzyskania wszystkich usług, które są dostępne i aktywne w systemie używamy polecenia `netstata` z parametrem `-a.` Opcja ta oferuje listę usług UDP i UTP bez względu na ich stan połączeń. Pod nagłówkiem UDP wyświetlany jest tylko adres lokalny i stan pola.

```
Debek:~# netstat -az
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 debek.promien.prz.r:ssh alfar:1040          ESTABLISHED
tcp        0      0 *:ftp                   *:*                    LISTEN
tcp        0      0 *:ssh                   *:*                    LISTEN
tcp        0      0 *:auth                  *:*                    LISTEN
```

```
tcp        0      0 *:pop3                  *:*                     LISTEN
tcp        0      0 *:smtp                  *:*                     LISTEN
tcp        0      0 *:1024                  *:*                     LISTEN
tcp        0      0 *:sunrpc                *:*                     LISTEN
udp        0      0 *:ntalk                 *:*
udp        0      0 *:talk                  *:*
udp        0      0 *:1024                  *:*
udp        0      0 *:856                   *:*
udp        0      0 *:sunrpc                *:*
raw        0      0 *:icmp                  *:*                     7
raw        0      0 *:tcp                   *:*                     7
```

- Wyświetlanie informacji o interfejsie

`Netstat` może zostać wykorzystane do uzyskania szczegółowych informacji na temat konfiguracji interfejsu oraz podstawowej liczby pakietów.
Opcja `-i` umożliwia uzyskanie listy wszystkich zdefiniowanych interfejsów w systemie.
Opcja `-n` jest wykorzystywana głównie do wyświetlenia adresów IP niż nazw hostów.

```
Debek:~# netstat -in
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 192.168.11.207:22      192.168.11.206:1040     ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node Path
unix  3      [ ]         DGRAM                    170    /dev/log
unix  1      [ W ]       STREAM     CONNECTED     306
unix  1      [ ]         STREAM     CONNECTED     305
unix  0      [ ]         DGRAM                    232
unix  0      [ ]         DGRAM                    204
unix  0      [ ]         DGRAM                    174
```

- Wyświetlanie informacji o trasowaniu

Tabela trasowania jest wykorzystywana przez system do określenia ścieżki, która ma być użyta w celu wysłania pakietów do poszczególnych hostów lub sieci. Do przejrzenia tej tabeli używamy opcji `-r`

```
Debek:~# netstat -I le11 r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
localnet        *               255.255.255.0   U         0 0          0 eth0
default         192.168.11.1    0.0.0.0         UG        0 0          0 eth0
```

8.4 NARZĘDZIE PING

Polecenie `ping` oferuje dwie podstawowe funkcje:

- można wykorzystać je do sprawdzenia, czy podstawowy poziom łączności jest możliwy pomiędzy jednym lub więcej punktami końcowymi, czy dane urządzenie jest dostępne z systemu lokalnego i do wspomagania łączności testowej pomiędzy systemami
- może oferować podstawowe dane statyczne dotyczące eksploatacji sieci

Stwierdzenie dostępności systemu.
Dzięki narzędziu `ping` możemy sprawdzić ogólną dostępność dowolnego urządzenia TCP/IP. Dla przykładu możemy sprawdzić, czy host alfar jest osiągalny? Użyjemy następującego polecenia:

```
Debek:~# ping alfar
PING alfar (192.168.11.206): 56 data bytes
64 bytes from 192.168.11.206: icmp_seq=0 ttl=128 time=0.8 ms
64 bytes from 192.168.11.206: icmp_seq=1 ttl=128 time=0.4 ms
64 bytes from 192.168.11.206: icmp_seq=2 ttl=128 time=0.4 ms
64 bytes from 192.168.11.206: icmp_seq=3 ttl=128 time=0.4 ms

--- alfar ping statistics ---
```

9. Agenci zarządzania sieciowego

- Agent nadrzędny SUN
- Agent SNMP Sun
- Agent SNMP UCP

Wszyscy wyżej wymienieni agenci oferują kilka takich samych usług:
- pomagają odzyskiwać i ustawiać obiekty MIB
- wspomagają standardowe, podstawowe czynności SNMP
- zapewniają kilka wyspecjalizowanych funkcji takich jak zarządzanie dyskiem i pamięcią

Działania każdego z agentów można podzielić na kilka kategorii:
- Wspomaganie MIB-II
- Zarządzanie obiektami
- Zdalna konfiguracja
- Rozszerzone wsparcie MIB
- Elastyczne pliki konfiguracyjne
- Obiekty eksploatacyjne
- Działania dopuszczające modyfikacje
- Zarządzanie dyskiem/pamięcią
- Wsparcie agenta nadrzędnego
- Rozszerzalność

9.1 Agent nadrzędny Sun

Sun oferuje agenta zwanego `snmpdx`, który jest wykorzystywany do zapewnienia zgodności agenta podrzędnego i nadrzędnego w przypadku jednego lub więcej agentów podrzędnych.
Agent `snmpdx` może zainstalować wiele wielu agentów podrzędnych w tym samym systemie bez żadnego konfliktu portowego. Agent nadrzędny wspiera zbiór obiektów MIB, które opisują agenta podrzędnego skonfigurowanego przez agenta nadrzędnego. [1]

9.2 Agent SNMP Sun

Agent ten zwany również `mibiisa` wspiera SNMPv1 i zapewnia dostęp w formie odczytu i zapisu do obiektów Sun i MIB-II. Oferuje możliwość zarządzania stacjami roboczymi Sun i serwerami z punktu widzenia operacyjnego i eksploatacyjnego. Wspiera on również standardowe grupy obiektów MIB-II o niektóre dodatkowe grupy zdefiniowane przez Sun. Agent ten może zostać wykorzystany do monitorowania niektórych bardziej krytycznych problemów zarządzania sieciowego.

Agent nadrzędny Sun oferuje następujące usługi:

- Otrzymywanie sieciowych danych statystycznych
- Otrzymywanie informacji o procesach systemowych
- Otrzymywanie informacji o pamięci systemu
- Wysyłanie określonych pułapek do określonych systemów

[1]

9.3 Agent SNMP UCD

Jest on również znany jako `snmpd` oferuje dużą listę usług związanych z SNMP. Wspiera on protokoły takie jak SNMPv1 i SNMPv2. Wspiera także MIB-II, UCD, MIB oraz inne bazy MIB. Oferuje możliwości zdalnego wykonywania poleceń, które zostały skonfigurowane na podstawie pewnych wydarzeń lub warunków. Może zostać wykorzystany do monitorowania pamięci systemu i informacji o procesach.

Agent UCD (snmpd) wspiera sieciowe dane statystyczne, bazy MIB, Host Resource MIB i UCD-SNMP MIB. Z punktu widzenia zarządzania sieciowego agent może być użyty do monitorowania krytycznych parametrów systemu operacyjnego i alarmowania, gdy zasoby osiągną poziom krytyczny. Za pomocą agenta możemy wykonać:

- Monitorowania zużycia przestrzeni dysku,
- Monitorowania obciążenia systemu,
- Monitorowania procesów systemowych,
- Monitorowania informacji o agentach i ich statusie,
- Wykonywania poleceń Unix i skryptów szufladowych,
- Zapewniania dostępu do kilku baz MIB,
- Zapewniania rozszerzeń MIB.

Uruchamianie systemu agenta

Aby agenci byli cały czas dostępni, powinni być uruchamiani w trakcie inicjalizacji systemu, lub gdy system jest restartowany. Agent Sun działa dzięki skryptowi inicjalizującemu i jest ustawiany podczas instalowania oprogramowania. Skrypt inicjalizujący *init.snmpdx* jest wykorzystywany do automatycznego uruchamiania agenta nadrzędnego. Gdy agent nadrzędny już funkcjonuje, uruchamia on agentów podrzędnych i domyślnie uruchamia agenta systemu `mibiisa` .
[1]

10 Funkcje monitorowania i administrowania

10.1 Polecenia UCD.

Pakiet UCD nie tylko zawiera solidnego i potężnego agenta SNMP, ale również wiele podręcznych narzędzi, które mogą być użyte do zarządzania urządzeniami sieciowymi. Narzędzia mogą zostać wykorzystane do budowania skryptów lub innych programów w celu wykonywania kompleksowych funkcji zarządzania sieciowego lub innych zadań.

Narzędzia UCD SNMP

-snmpbulkwalk -pobiera obiekty MIB za pomocą żądań grupowych,
-snmpdelta -monitoruje zmiany zmiennych SNMP,
-snmpget -pobiera jeden lub więcej obiektów MIB,
snmpgetnet -ciągle przemieszcza się po drzewie SNMP MIB i otrzymuje wszystkie wspierane obiekty,
-snmpnetstart –otrzymuje informacje o konfiguracji interfejsu agenta,
-snmpset –ustawia jeden lub więcej obiektów MIB na określoną wartość,
-snmpstatus –otrzynuje ważne informacje o obiektach MIB,
-snmotable –konwertuje obiekty MIB w bardziej znaczące informacje,
-snmptrap –wysyła wiadomości pułapki SNMP do jednego lub więcej programów zarządzania,
-snmptrapd –pobiera pułapki SNMP a sieci,
-snmpwalk –otrzymuje grupę powiązanych obiektów MIB.

Składnia narzędzi UCD:
Snmpcd wersja_protokołu [dodatkowe_opcje] nazwa_hosta obiekt_współdziałania [obiekt] [1]

Popularne opcje poleceń.

Narzędzia UCD mają pewną liczbę wspólnych argumentów wiersza zlecenia. Posiadanie podstawowego zestawu opcji ułatwia zapamiętywanie i używanie narzędzi. Argumenty oferowane przez wszystkie narzędzia są podzielone na dwie kategorie :
-opcje operacyjne,
-opcje wyświetlania. [1]

10.2 Snmpbulkwalk.

Narzędzie snmpbulkwalk jest wykorzystywane do komunikowania się z elementami sieci przy użyciu żądań SNMPv2 SNMP GET BULK. Podobnie jak narzędzie snmpwalk, snmpbulwalk przemieszcza się po drzewie MIB, dopóki nie dotrze do końca MIB lub nie pojawi się jakiś błąd.
Składnia polecenia:
Snmpbulwalk –v 2c cisco-gwl public
Polecenie snmpbulwalk współpracuje tylko z agentami SNMPv2, ponieważ funkcja GET BULK nie była wdrażana we wcześniejszych wersjach. [1]

10.3 Snmpdelta.

Polecenie snmpdelta gromadzi zmiany dokonane w wartościach liczb całkowitych MIB pochodzące od elementu agenta SNMP. To polecenie monitoruje podany obiekt całkowitoliczbowy i wyświetla zmiany dokonywane w obiekcie, które pojawiły się na przestrzeni czasu.

Składnia polecenia:
Snmpdelta –R remote –gw public –m ifOutOcttets.1 [1]

10.4 Snmpget.
Polecenie snmpget jest wykonywane do pobierania informacji z elementów agenta SNMP. Używa ono żądania SNMP GET wraz a jedną lub więcej nazw w pełni wykwalifikowanych obiektów w formie argumentów i przywraca ich wartości.
Składnia polecenia:
Snmpget [zwykłe argumenty] obiekt MIB [obiekt MIB] [1]

10.5 Snmpgetnext.
Polecenie snmpgetnext jest używane do pobierania jednego lub więcej obiektów MIB przy użyciu żądań SNMP GET Next. Dla każdego obiektu podanego w wiersza zlecenia snmpgetnext otrzymuje następnie leksykograficzny obiekt MIB, znaleziony na drzewie MIB.
Składnia polecenia:
Snmpgetnext 10.0.2.220 public system.sysContact.0 [1]

10.6 Snmpnetstat.
Polecanie snmpnetstat jest podobne do polecenia Unix netstat i oferuje wiele tych samych podstawowych informacji na temat załączonych interfejsów urządzeń i tabel trasowania. Ważną cechą tego programu użytkowego jest to, że ułatwia on uzyskiwanie informacji o interfejsach dla jakiegokolwiek urządzenia SNMP(routery, węzły komutacyjne, sondy monitorujące i inne).

Składnia polecenia:
Snmpnetstat –i cisco –gw3 public [1]

10.7 Snmpset.
Polecenie snmpset jest jednym z najbardziej użytecznych i potężnych poleceń w pakiecie UCD. To narzędzie jest wykorzystywane do poprawiania modyfikowalnych obiektów agentów MIB.
Możliwość poprawiania obiektów MIB jest ogromna, ponieważ powoduje ona zmiany w konfiguracji i stanie operacyjnym zarządzanego agenta.
Podstawowa składnia polecenia:
Snmpset [zwykły argument] ID obiektu MIB typ wartości [ID obiektu MIB typ wartości]
Powody używania polecenia snmpset:
-Odłączenie lub połączenie interfejsu sieciowego,
-Zaktualizowane urządzenia a nowymi informacjami o administracji ,
-Skasowanie pewnych liczników,
-Zrestartowanie urządzenia lub agenta,
-Zmodyfikowanie niektórych parametrów konfiguracyjnych. [1]

10.8 Snmpstatus.
Polecenie snmpstastus otrzymuje ważne informacje od elementu sieci SNMP używającego polecenia snmpget.
Snmpstatus jest używane w odniesieniu do urządzenia, wyświetla następujące informacje:

-adres IP urządzenia,
-obiekt MIB sysDescr,
-obiekt MIB sysUpTime,
-liczbę pakietów otrzymanych i transmitowanych we wszystkich aktywnych interfejsach
-liczbę pakietów IP otrzymanych i transmitowanych,
-liczbę aktywnych interfejsów,
-liczbę interfejsów, które zostały odłączone.
Składnia polecenia:
Snmpstatus host znak_współdziałania [1]

10.9 Snmptable.
Polecenie snmptable zapewnia możliwość otrzymania pełnej tabeli MIB przy użyciu żądania SNMP GETNEXT.
Podstawowym celem tego polecenia jest danie użytkownikowi możliwości wyświetlania tabel SNMP i importowania danych do innych programów w celu dalszych modyfikacji i tworzenia raportów.
Składnia polecenia:
Snmptable [zwykłe opcje] host znak_współdziałania ID_tabeli [1]

10.10 Snmptest.
Polecenie snmptest oferuje prostą funkcję, która ułatwia komunikowanie się z elementami sieci przy użyciu SNMP.
Oprogramowanie zapewnia trzy tryby operacyjne:
-get,
-getnext,
-set. [1]

10.11 Snmptranslate.
Narzędzie snmptranslate jest używane do tłumaczenia obiektów SNMP MIB na bardziej czytelny tekst. Kiedy to polecenie jest uruchamiane, przetłumaczy ono obiekt MIB na wartość SMI lub formę symboliczną. Podstawowym celem wykorzystania polecenia jest pomoc przy wyświetlaniu pełnych charakterystyk obiektów MIB, bez konieczności zaglądania do odpowiednich plików definicji MIB.
Składnia polecenia:
Snmptranslate –d system.sysDesct [1]

10.12 Snmptrap.
Polecenie snmptrap będzie emitowało pułapkę SNMP do wyznaczonego programu zarządzania SNMP. To narzędzie jest bardzo użyteczne, jeśli jest osadzone w skrypcie szufladowym lub innym programie służącym do wysyłania pułapek do wielu programów zarządzania siecią SNMP.
Podstawowa składnia polecenia:
Snmptrap –v 1 [argumenty polecenia] OID-przedsiębiorstwa agent ogólna-pułapka specyficzna-pułapka czas-pracy [ID obiektu typ wartości] [1]

10.13 Snmptrapd.
Polecenie snmptrapd otrzymuje i loguje pułapki SNMP, które są wysyłane w porcie 162, są również logowane w funkcji Unix syslog lub wyświetlane na terminalu.

Polecenie to musi być uruchamiane przez uprzywilejowanego użytkownika, ponieważ jego działanie jest oparte na zastrzeżonym porcie systemu. Wykonanie bez żadnych opcji spowoduje umieszczenie go w tle i odłączenie od warstwy wywołującej.

10.14 Snmpwalk.

Polecenie snmpwalk przemieszcza się po drzewie MIB agenta, używając żądania SNMP GETNEXT. Zmienna obiektu może być podana w wierszu zlecenia w celu określenia, w której części przestrzeni MIB powinno się rozpocząć przeszukiwanie.
Składnia polecenia:
Snmpwalk monet public [1]

10.15 Snmpconf.

Polecenie to oferuje możliwość konfigurowania urządzeń przy użyciu polecenia snmpset zgodnie z obiektami MIB zdefiniowanymi w pliku konfiguracyjnym. Plik konfiguracyjny może zawierać listę obiektów MIB i wartości, które zostaną ustawione w stosunku do urządzenia SNMP. To narzędzie zapewnia zautomatyzowany mechanizm do stosowania informacji o standardowej konfiguracji.
Snmpconf zapewnia podstawowe sprawdzanie błędów, a kiedy błąd się pojaw, wyświetla odpowiednią informację.
Do tego celu używamy plików konfiguracyjnych.
Plik taki może zawierać pozycje obiektów MIB, komentarze i puste wersy.
Jeżeli plik konfiguracyjny zawiera wymagane obiekty MIB i wartości, może on być użyty w stosunku do urządzenia SNMP. Podstawowa składnia polecenia:
Snmpconf nazwa_hosta znak-współdziałania plik konfiguracyjny [1]

Literatura:

[1.] Steve Maxwell „UNIX –Narzędzia do zarządzania siecią”
[2.] „Vademecum teleinformatyka”

Przykłady zostały wykonane na systemie operacyjnym LINUX-DEBIAN

[3] Linki do stron:

http://hip.ipadmin.info/own/mrtg/mrtg_referat.htm

http://www.man.rzeszow.pl/docs/ip/intro.htm