Weryfikacja łączności

17.4.1

Testowanie łączność za pomocą ping

Niezależnie od tego, czy Twoja sieć jest mała i nowa, czy też skalujesz istniejącą sieć, zawsze będziesz chciał sprawdzić, czy Twoje komponenty są prawidłowo połączone ze sobą i z Internetem. W tym temacie omówiono niektóre narzędzia, których można użyć do sprawdzenia połączenia w sieci.

Polecenie **ping** jest najskuteczniejszym sposobem, aby szybko przetestować łączność warstwy 3 między źródłowym i docelowym adresem IP. Polecenie wyświetla również różne statystyki czasu wymiany w obie strony.

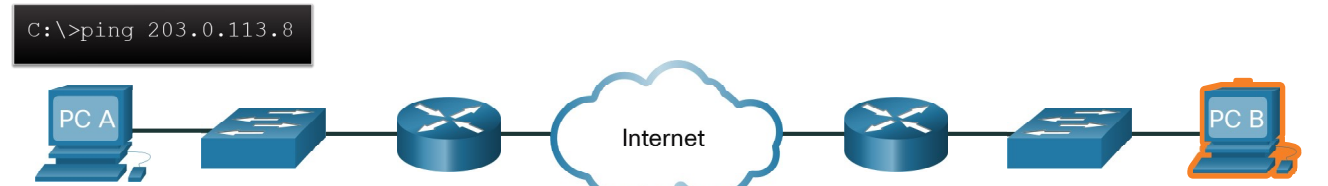
W szczególności polecenie **ping** wykorzystuje komunikaty echa protokołu ICMP (ICMP typu 8) i odpowiedzi echa (ICMP typu 0). Polecenie **ping** jest dostępne w większości systemów operacyjnych, w tym Windows, Linux, MacOS i Cisco IOS.

Na hoście systemu Windows 10 polecenie **ping** wysyła cztery kolejne komunikaty echo ICMP i oczekuje czterech kolejnych odpowiedzi echa ICMP od odbiorcy.

Na przykład załóżmy, że PC A uruchamia ping do PC B. Jak pokazano na rysunku, komputer A Windows host wysyła cztery kolejne komunikaty echo ICMP do PC B (tj. 10.1.1.10).

Źródłowy adres IP	Docelowy adres IP	ICMP
192.168.10.10	10.1.1.10	Echo

Host docelowy odbiera i przetwarza echa ICMP. Jak pokazano na rysunku, PC B odpowiada wysyłając cztery komunikaty ICMP echo replay do PC A.



Źródłowy IP	Docelowy IP	ICMP
10.1.1.10	192.168.10.10	Odpowiedzi Echo

Jak pokazano na wyjściu poleceń, PC A otrzymał odpowiedzi echo od PC-B sprawdzające połączenie sieciowe warstwy 3.

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time=47ms TTL=51
Reply from 10.1.1.10: bytes=32 time=60ms TTL=51
Reply from 10.1.1.10: bytes=32 time=53ms TTL=51
Reply from 10.1.1.10: bytes=32 time=50ms TTL=51
Ping statistics for 10.1.1.10:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 47ms, Maximum = 60ms, Average = 52ms
C:\Users\PC-A>
```

Wyjście sprawdza łączność warstwy 3 pomiędzy PC A i PC B.

Wyjście polecenia **ping** Cisco IOS różni się od tego z hosta systemu Windows. Na przykład, ping IOS wysyła pięć komunikatów echo ICMP, jak pokazano na wyjściu.

```
R1# ping 10.1.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
R1#
```

Zwróć uwagę na znaki wyjściowe **!!!!!**. Polecenie **ping** IOS wyświetla wskaźnik dla każdego otrzymanego echa ICMP. Tabela zawiera listę najczęstszych znaków wyjściowych z polecenia **ping**.

Wskaźniki ping IOS

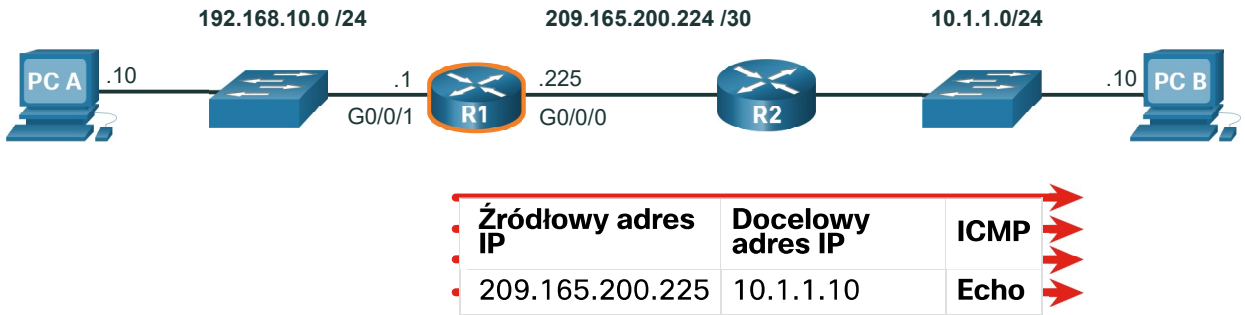
Element	Opis
!	• Wykrzyknik wskazuje pomyślne otrzymanie odpowiedzi komunikatu echa . • Sprawdza połączenie warstwy 3 między nadawcą i odbiorcą.
.	• Kropka oznacza, że upłynął czas oczekiwania na wiadomość odpowiedzi echa. • Wskazuje to na problem z łącznością występujący gdzieś wzdłuż ścieżki.
U	• Wielkie **U** wskazuje router na trasie, który odpowiedział komunikatem błędu ICMP Type 3 cel nieosiągalny. • Możliwe przyczyny to, że router nie zna kierunku do sieci docelowej lub nie mógł znaleźć hosta w sieci docelowej.

Uwaga: Inne możliwe odpowiedzi na ping to Q, M,? , lub &. Jednak ich znaczenie jest poza zakresem tego modułu.

17.4.2

Rozszerzona wersja polecenia ping

Standardowy **ping** wykorzystuje adres IP interfejsu znajdującego się najbliżej sieci docelowej jako źródło komunikatów **ping**. Źródłowy adres IP polecenia **ping 10.1.1.10** na R1 byłby adresem interfejsu G0/0/0 (tj. 209.165.200.225), jak zilustrowano w przykładzie.



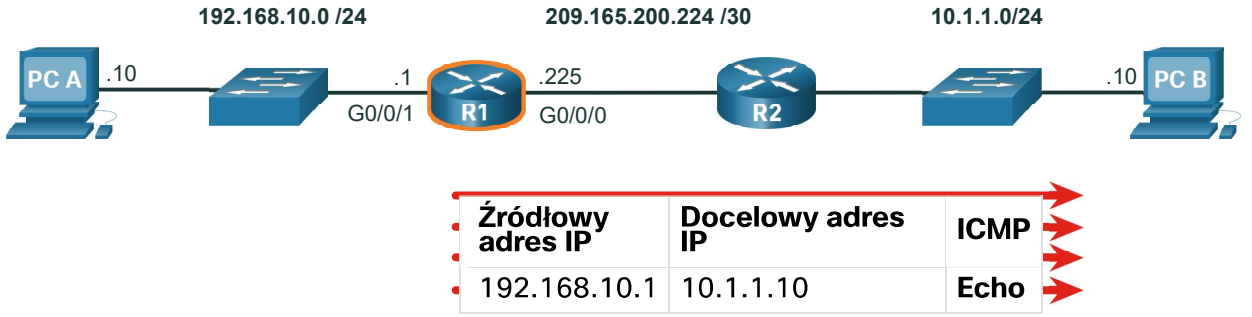
System operacyjny Cisco IOS oferuje rozszerzoną wersję komendy **ping**. Tryb ten umożliwia użytkownikowi tworzenie specjalnego typu testów ping poprzez dostosowanie parametrów związanych z działaniem polecenia.

Aby wejść w ten tryb, należy wpisać **ping** w wierszu poleceń CLI w trybie uprzywilejowanym EXEC bez określania docelowego adresu IP. Następnie otrzymasz kilka monitów, aby dostosować rozszerzony **ping**.

Uwaga: Naciśniecie **Enter** powoduje akceptację wskazanej domyślnej wartości.

Załóżmy na przykład, że chcesz przetestować łączność z R1 LAN (tj. 192.168.10.0/24) do 10.1.1.0 LAN. Można to sprawdzić z komputera A. Jednak rozszerzony **ping** może być użyty na R1, aby określić inny adres źródłowy.

Jak ilustrowano w przykładzie, źródłowy adres IP rozszerzonego polecenia **ping** na R1 można skonfigurować tak, aby używał adresu IP interfejsu G0/0/1 (tj. 192.168.10.1).



Poniższe polecenie wyjściowe konfiguruje rozszerzony **ping** na R1 i określa źródłowy adres IP, który ma być adresem interfejsu G0/0/1 (tj. 192.168.10.1).

```
R1# ping
Protocol [ip]:
Target IP address: 10.1.1.10
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Ingress ping [n]:
Source address or interface: 192.168.10.1
DSCP Value [0]:
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0x0000ABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.10.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
R1#
```

Uwaga: Polecenie **ping ipv6** służy do rozszerzonych testów ping IPv6.

17.4.3

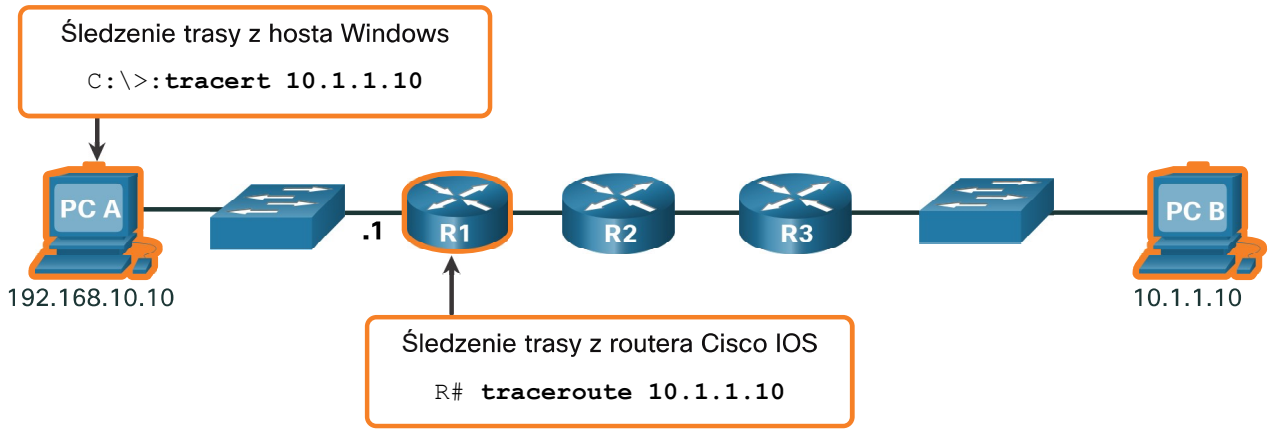
Sprawdzanie połączeń za pomocą traceroute

Polecenie **ping** jest przydatne, aby szybko określić, czy występuje problem z łącznością warstwy 3. Jednak nie identyfikuje, gdzie problem znajduje się wzdłuż ścieżki.

Traceroute może pomóc zlokalizować obszary problemowe warstwy 3 w sieci. Mechanizm śledzenia trasy zwraca listę adresów kolejnych przeskoków na trasie pakietu. Można go użyć do zidentyfikowania punktu na ścieżce, w którym można znaleźć problem.

Składnia polecenia trace jest różna między systemami operacyjnymi, jak pokazano na rysunku.

Polecenia Windows i Cisco IOS śledzenia ścieżki



Poniżej znajduje się przykładowe wyjście polecenia **tracert** na hoście systemu Windows 10.

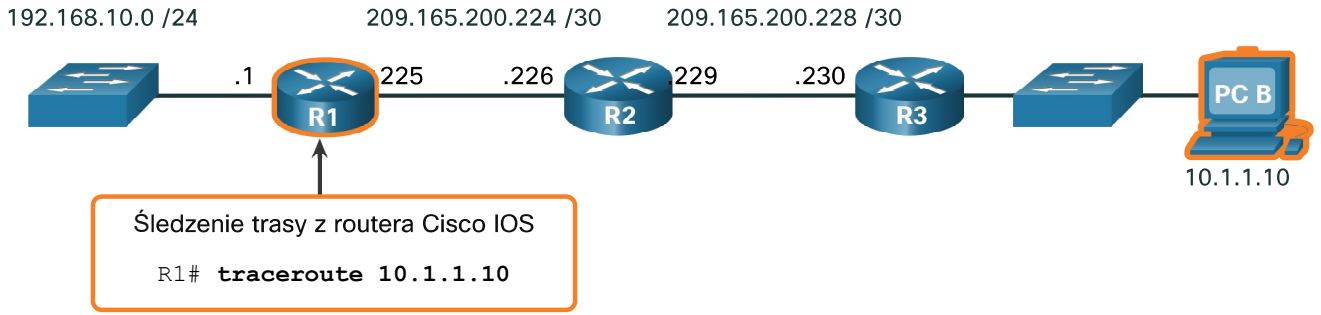
```
C:\Users\PC-A> tracert 10.1.1.10
```

```
Tracing route to 10.1.10 over a maximum of 30 hops:
  1     2 ms     2 ms     2 ms  192.168.10.1
  2     *        *        *     Request timed out.
  3     *        *        *     Request timed out.
  4     *        *        *     Request timed out.
^C
C:\Users\PC-A>
```

Uwaga: Użyj **Ctrl-C** aby przerwać **tracert** w Windows.

Jedyną udaną reakcją była brama R1. Upłynął limit czasu żądań śledzenia do następnego przeskoku, jak wskazano gwiazdką (*), co oznacza, że router następnego przeskoku nie odpowiedział. Przekroczone limity czasu wskazują, że w sieci znajduje się błąd poza siecią LAN lub że routery te zostały skonfigurowane tak, aby nie odpowiadały na żądania echa użyte w tracert. W tym przykładzie pojawia się problem między R1 i R2.

Wyjście polecenia **traceroute** Cisco IOS różni się od polecenia **tracert** Windows. Zapoznaj się z przykładową, następującą topologią.



Poniżej znajduje się przykładowy wynik polecenia traceroute z R1.

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
  1 209.165.200.226 1 msec 0 msec 1 msec
  2 209.165.200.230 1 msec 0 msec 1 msec
```

```
  3 10.1.1.10 1 msec 0 msec
R1#
```

W tym przykładzie, polecenie potwierdziło, że może z powodzeniem dotrzeć do PC B.

Limity czasu wskazują na potencjalny problem. Na przykład, jeśli komputer 10.1.1.10 nie był dostępny, polecenie **traceroute** wyświetli następujący komunikat.

```
R1# traceroute 10.1.1.10
Type escape sequence to abort.
Tracing the route to 10.1.1.10
VRF info: (vrf in name/id, vrf out name/id)
  1 209.165.200.226 1 msec 0 msec 1 msec
  2 209.165.200.230 1 msec 0 msec 1 msec
  3 * * *
  4 * * *
  5 *
```

Użyj **Ctrl-Shift-6** , aby przerwać **traceroute** w Cisco IOS.

Uwaga: Implementacja systemu Windows traceroute (tracert) wysyła żądania echa ICMP. Cisco IOS i Linux używają UDP z nieprawidłowym numerem portu. Ostateczny cel zwróci komunikat port nieosiągalny ICMP.

17.4.4

Rozszerzone polecenie traceroute

Podobnie jak rozszerzone polecenie **ping**, istnieje również rozszerzone polecenia **traceroute**. Pozwala to administratorowi dostosować parametry związane z działaniem polecenia. Jest to pomocne w zlokalizowaniu problemu podczas rozwiązywania problemów z pętlami routingu, dokładnego określania routera następnego przeskoku lub określania, gdzie pakiety są odrzucane lub blokowane przez router lub zaporę.

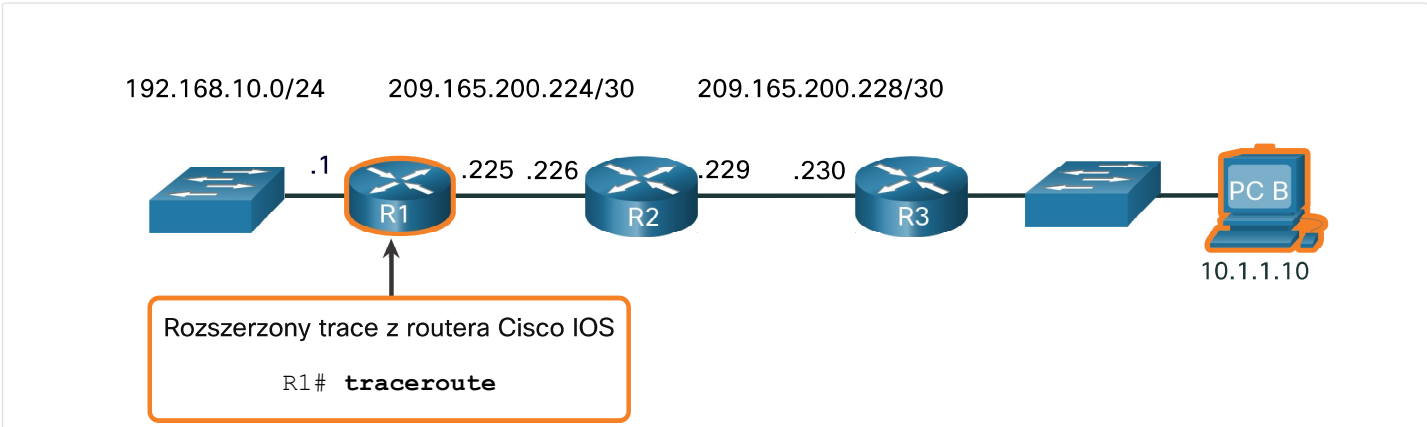
Polecenie **tracert** Windows umożliwia wprowadzanie kilku parametrów za pomocą opcji w wierszu poleceń. Jednak nie jest tak realizowany jak rozszerzone polecenia traceroute IOS. Poniższy wynik wyświetla dostępne opcje dla polecenia **tracert** Windows.

```
C:\Users\PC-A> tracert /?
Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
               [-R] [-S srcaddr] [-4] [-6] target_name
Options:
    -d                 Do not resolve addresses to hostnames.
    -h maximum_hops    Maximum number of hops to search for target.
    -j host-list       Loose source route along host-list (IPv4-only).
    -w timeout         Wait timeout milliseconds for each reply.
    -R                 Trace round-trip path (IPv6-only).
    -S srcaddr         Source address to use (IPv6-only).
    -4                 Force using IPv4.
    -6                 Force using IPv6.
C:\Users\PC-A>
```

Opcja rozszerzona **traceroute** Cisco IOS umożliwia użytkownikowi tworzenie specjalnego typu śledzenia poprzez dostosowanie parametrów związanych z działaniem polecenia. Aby wejść w ten tryb, należy wpisać **traceroute** w wierszu poleceń CLI w trybie uprzywilejowanym EXEC bez określania docelowego adresu IP. IOS poprowadzi Cię przez opcje poleceń, przedstawiając szereg monitów związanych z ustawieniem wszystkich różnych parametrów.

Uwaga:Naciśnięcie **Enter** powoduje akceptację wskazanej domyślnej wartości.

Na przykład załóżmy, że chcesz przetestować łączność z komputerem B z R1 LAN. Chociaż można to sprawdzić z PC A, można skonfigurować rozszerzony **traceroute** na R1, aby określić inny adres źródłowy.



Jak pokazano w przykładzie, źródłowy adres IP rozszerzonego polecenia **traceroute** na R1 może być skonfigurowany do

korzystania z adresu IP interfejsu R1 LAN (tj. 192.168.10.1).

```
R1# traceroute
Protocol [ip]:
Target IP address: 10.1.1.10
Ingress traceroute [n]:
Source address: 192.168.10.1
DSCP Value [0]:
Numeric display [n]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Type escape sequence to abort.
Tracing the route to 192.168.10.10
VRF info: (vrf in name/id, vrf out name/id)
  1 209.165.200.226 1 msec 1 msec 1 msec
  2 209.165.200.230 0 msec 1 msec 0 msec
  3  *
    10.1.1.10 2 msec 2 msec
R1#
```

17.4.5

Wyznaczenie stanu odniesienia sieci

Jedną z najbardziej efektywnych metod monitorowania i rozwiązywania problemów z wydajnością sieci jest wyznaczenie jej charakterystyki bazowej. Stworzenie charakterystyki bazowej sieci wymaga czasu. Pomiar wydajności i obciążenia sieci w różnych chwilach czasu pomaga w utworzeniu lepszego obrazu ogólnej wydajności sieci.

Dane wyjściowe pochodzące z poleceń sieciowych wnoszą dane do stanu odniesienia sieci. Jedną z metod tworzenia stanu odniesienia jest skopiowanie i wklejenie wyników wykonanego polecenia **ping**, **trace** lub innych odpowiednich poleceń do pliku tekstowego. Pliki powinny zawierać informacje o czasie i być archiwizowane w celu późniejszego wydobycia z nich danych i

porównania.

Wśród pozycji do rozważenia są komunikaty o błędach i czasy odpowiedzi od hostów. Jeśli zanotowano znaczący wzrost w czasach odpowiedzi, może to oznaczać, że istnieje problem z opóźnieniem.

Na przykład, następujący wynik polecenia **ping** został przechwycony i wklejony do pliku tekstowego.

19 września 2019 o 10:18:21

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time<1ms TTL=64
Reply from 10.1.1.10: bytes=32 time<1ms TTL=64
Reply from 10.1.1.10: bytes=32 time<1ms TTL=64
Reply from 10.1.1.10: bytes=32 time<1ms TTL=64
Ping statistics for 10.1.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\PC-A>
```

Zauważ, że czasy **ping** wymiany w obie strony są mniejsze niż 1 ms.

Miesiąc później ping jest powtarzany i przechwytywany.

19 września 2019 o 10:18:21

```
C:\Users\PC-A> ping 10.1.1.10
Pinging 10.1.1.10 with 32 bytes of data:
Reply from 10.1.1.10: bytes=32 time=50ms TTL=64
Reply from 10.1.1.10: bytes=32 time=49ms TTL=64
Reply from 10.1.1.10: bytes=32 time=46ms TTL=64
Reply from 10.1.1.10: bytes=32 time=47ms TTL=64
Ping statistics for 10.1.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 46ms, Maximum = 50ms, Average = 48ms
C:\Users\PC-A>
```

Zauważ tym razem, że czasy **ping** wymiany w obie strony są znacznie dłuższe i wskazują na potencjalny problem.

Sieci firmowe powinny posiadać dokładnie określoną charakterystykę bazową – dużo bardziej szczegółowo niż opisuje to szkolenie. Dostępne są profesjonalne narzędzia do przechowywania i utrzymywania danych charakterystyki bazowej. W czasie tego szkolenia poznamy podstawowe techniki i omówimy cele charakterystyki bazowej.

Najlepsze praktyki Cisco w zakresie procesów bazowych można znaleźć, przeszukując Internet w poszukiwaniu „Bazeline Process Best Practice”.

17.4.6

Laboratorium - Testowanie opóźnienia sieci za pomocą polecenia ping i traceroute

Celem tego ćwiczenia jest realizacja następujących zadań:

- Część 1: Wykorzystanie polecenia ping do dokumentowania opóźnień w sieci
- Część 2: Używanie polecenia traceroute do dokumentowania opóźnień w sieci

Testowanie opóźnienia sieci za pomocą polecenia ping i traceroute

17.3 Skalowanie do większej sieci

17.5 Polecenia na komputerze i w systemie IOS