ANHANG P

Point-to-Point-WANs implementieren

Standleitungs-WANs (auch als serielle Verbindungen bezeichnet) erfordern weitaus weniger Überlegungen als viele andere Themen – zumindest in dem Maße, wie es für die CCENT- und CCNA R&S-Prüfungen erforderlich ist. Diese Einfachheit erlaubt es, Standleitungen lediglich in der ICND1-Prüfung zu berücksichtigen und als Unterbereich des IP-Routings zu behandeln.

In diesem Kapitel beschäftigen wir uns nun eingehender als bisher mit den Standleitungs-WANs. Als Grundlage wiederholen wir kurz das Konzept für Standleitungen aus dem ICND1-Buch und wenden es dann auf andere Konzepte an. Wichtiger sind die in diesem Kapitel besprochenen Schritte für Konfiguration, Überprüfung und Troubleshooting von Standleitungen, die das bekannte Data-Link-Protokoll HDLC (High-Level Data Link Control) und das Point-to-Point Protocol (PPP) verwenden.

In diesem Anhang werden wir die Themen in drei Hauptabschnitten vorstellen. Im ersten sehen wir uns Standleitungs-WANs an, die mit HDLC arbeiten, und wir erörtern die physischen Verbindungen und deren Details sowie die Konfiguration von HDLC (und zugehörige Themen). Der zweite Hauptabschnitt widmet sich PPP. Dieses alternative Data-Link-Protokoll können Sie statt HDLC verwenden, wobei wir uns schwerpunktmäßig auf Konzept und Konfiguration konzentrieren. Im dritten Abschnitt schließlich wird es um die Ursachen von Problemen bei seriellen Verbindungen gehen und darum, wie man sie findet.

Grundlagenthemen

Standleitungs-WANs mit HDLC

Ein physisches Standleitungs-WAN funktioniert ähnlich wie ein Ethernet-Crossover-Kabel, das zwei Router miteinander verbindet, aber es gibt keine Begrenzungen hinsichtlich der Entfernung. Wie in Abbildung P.1 gezeigt, kann jeder Router jederzeit senden (Vollduplexbetrieb). Die Geschwindigkeit ist ebenfalls symmetrisch, d. h., beide Router können Bits mit derselben Datenrate senden.

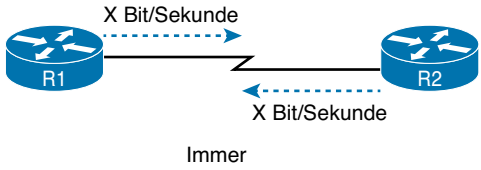


Abbildung P.1 Standleitung: gleiche Datenrate in beiden Richtungen, immer eingeschaltet

Obwohl die Standleitung eine Übertragungseinrichtung des Physical Layer ist, müssen Router für das Versenden von Bits über die WAN-Verbindung zusätzlich ein Data-Link-Protokoll einsetzen. Das sollte sich mittlerweile vertraut anhören: Router empfangen Frames über LAN-Interfaces und entkapseln daraus dann das Network-Layer-Paket. Vor der Weiterleitung des Pakets kapselt der Router es dann wieder in einem WAN-Data-Link-Protokoll wie HDLC (Abbildung P.2, Schritt 2).

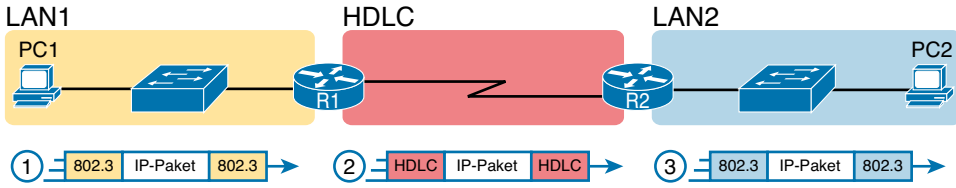


Abbildung P.2 Wie Router HDLC nutzen, um Pakete zu kapseln

In den ersten beiden Abbildungen werden einige Details der Layer 1 und 2 von Standleitungs-WANs wiederholt. Im ersten Hauptabschnitt dieses Anhangs beginnen wir noch einmal mit den Ausführungen zu diesen Details – zuerst zu Layer 1, dann zu Layer 2. Der Abschnitt endet mit Erläuterungen zur HDLC-Konfiguration.

Layer-1-Standleitungen

Standleitungen gibt es schon sehr lange – ungefähr zwanzig Jahre länger als LANs. Und auch heute existieren sie immer noch als WAN-Dienst.

Als Folge dieser langen Marktpräsenz gibt es in der Netzwerktechnik eine ganze Reihe unterschiedlicher Benennungen. So stößt man auch auf die Bezeichnung *Mietleitung* (engl. *Leased Line*), aus der hervorgeht, dass die Standleitung nicht dem Benutzer gehört, sondern dieser regelmäßig eine Gebühr für ihre Benutzung entrichtet. Oft wird dieser Dienst bei einem Telekommunikationsunternehmen geleast. Außerdem wird heute meistens der allgemeine Begriff *Serviceprovider* verwendet. Er bezeichnet ein Unternehmen, das eine bestimmte Form der

WAN-Konnektivität anbietet, z. B. Internetdienste. In Tabelle P.1 werden einige dieser Bezeichnungen aufgelistet, damit Sie die verschiedenen Begrifflichkeiten lernen, denen Sie als Netzwerktechniker in der Praxis begegnen könnten.

Tabelle P.1 Unterschiedliche Bezeichnungen für Standleitungen

Name	Bedeutung oder Referenz
Mietleitung, geleaste Leitung	Die Termini *Leitung* und *Verbindung* werden in der Sprache der Telekommunikation synonym verwendet. Der Begriff Standleitung verweist auf die Tatsache, dass diese Leitung jederzeit verfügbar ist, die Verbindung also nicht extra hergestellt werden muss.
Serieller Link, serielle Verbindung	Wiederum werden die Wörter *Leitung* und *Verbindung* in diesem Kontext häufig synonym verwendet. *Seriell* bezieht sich auf die Tatsache, dass die Bits seriell (also nacheinander) übertragen werden und die Router serielle Interfaces verwenden.
Point-to-Point-Leitung, Point-to-Point-Link	Bezeichnet die Tatsache, dass sich die Topologie auf genau zwei Punkte erstreckt. (Einige ältere Standleitungsmethoden ließen auch mehr als zwei Geräte zu.)
T1	Ein spezieller Standleitungstyp mit einer Datenübertragungsrate von 1,544 Megabit pro Sekunde (Mbit/s)
WAN-Leitung, WAN-Link	Beide Begriffe sind eher generisch, stehen also eigentlich nicht im Zusammenhang mit einer bestimmten Technologie.

Physische Komponenten von Standleitungen

Für den Aufbau einer Standleitung muss ein physischer Pfad zwischen den beiden Routern an den Enden der Leitung vorhanden sein. Die physische Verkabelung muss an den Gebäuden austreten, in denen die Router sich befinden. Zudem muss der Serviceprovider ein Kabel mit zwei Leiterpaaren von einem Ende zum anderen installieren, wobei jeweils über ein Leitungspaar die Daten in eine Richtung gesendet werden (Vollduplex). Abbildung P.3 zeigt ein Beispiel, bei dem der Provider einige traditionelle CO-Switches für eine kurze Standleitung zwischen zwei Routern verwendet.

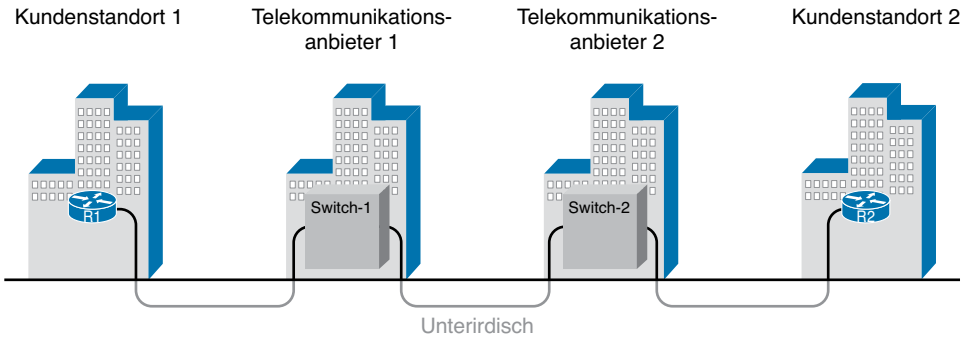


Abbildung P.3 Verkabelung einer kurzen Standleitung über einen Provider (Beispiel)

Die Details in der Mitte von Abbildung P.3 zeigen alles, was Sie wahrscheinlich jemals über Standleitungs-WANs wissen müssen (zumindest aus der Sicht von Unternehmenskunden). Häufiger jedoch denken die meisten Netzwerktechniker über Standleitungen eher aus Sicht von Abbildung P.4 nach. Darin sind einige zentrale Komponenten und Begriffe für Geräte an den Enden der Standleitungen aufgeführt:

- **Customer Premises Equipment (CPE):** Bezeichnet Geräte, die sich am kundenseitigen Ende der Verbindung befinden.
- **Channel Service Unit/Data Service Unit (CSU/DSU):** Ein solches Gerät sorgt für eine Funktion namens *Taktung*, wodurch physisch die Datenrate und das Timing gesteuert werden, mit denen das serielle Router-Interface Daten über die serielle Leitung sendet und empfängt.
- **Serielles Kabel:** Mit diesem kurzen Kabel wird die CSU mit dem seriellen Interface des Routers verbunden.

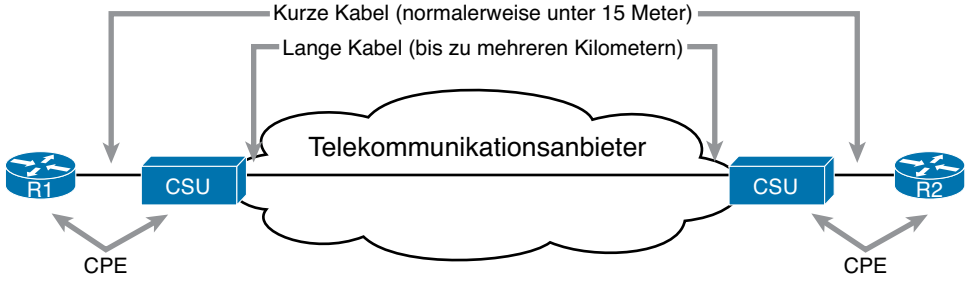


Abbildung P.4 Point-to-Point-Standleitung: Komponenten und Terminologie

Zum CPE gehören verschiedene, separat bestellbare Teile. Bei Verwendung einer externen CSU/DSU muss die Verbindung über ein serielles Kabel mit dem seriellen Interface des Routers hergestellt werden. Diese seriellen Interfaces sind normalerweise Bestandteil einer austauschbaren Steckkarte im Router. Diese Karten heißen WIC (WAN Interface Card), Highspeed WIC (HWIC) oder Netzwerkinterfacemodule (NIM). Die meisten seriellen Interfaces weisen einen physischen Steckverbinder bestimmter Form und Größe – den sogenannten *Smart Serial-Steckverbinder* – auf, während die CSU in der Regel mit einem anderen Steckverbinder ausgestattet ist. Bei der Installation der Standleitung muss der Netzwerktechniker den korrekten Kabeltyp wählen, damit die Steckverbindungen vom WIC am einen Ende zum CSU/DSU am anderen passen. Abbildung P.5 zeigt einen Typ für ein serielles Kabel: Der Smart Serial-Steckverbinder befindet sich auf der linken Seite, der verbreitete V.35-Steckverbinder auf der rechten. Die Abbildung zeigt eine seitliche Gesamtansicht des Kabels sowie Draufsichten auf die Steckverbinder an den Kabelenden.

Heute werden bei Standleitungen häufig Cisco-WICs mit einer integrierten CSU/DSU verwendet. Die WIC-Hardware bietet dabei die gleichen Funktionen wie eine CSU/DSU, weswegen keine externe CSU/DSU benötigt wird. Verglichen mit Abbildung P.4 sind die externe CSU/DSU und das serielle Kabel an beiden Enden nicht nötig, da das Kabel des Serviceproviders direkt mit der WIC verbunden ist.

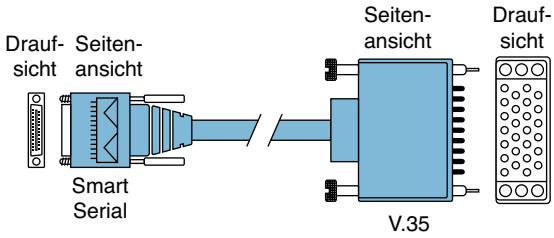


Abbildung P.5 Serielle Kabel zwischen einer CSU und einem Router

Abbildung P.6 zeigt einen Router mit zwei NIM-Steckplätzen. Jeder Steckplatz ist hier hinter einer Blende verborgen und es ist keine NIM-Karte installiert. Im Vordergrund der Abbildung sehen wir ein NIM mit zwei seriellen Ports mit Smart Serial-Interface. Das Kabelende auf der linken Seite von Abbildung P.5 könnte an einen dieser NIM-Anschlüsse in Abbildung P.6 angeschlossen werden.

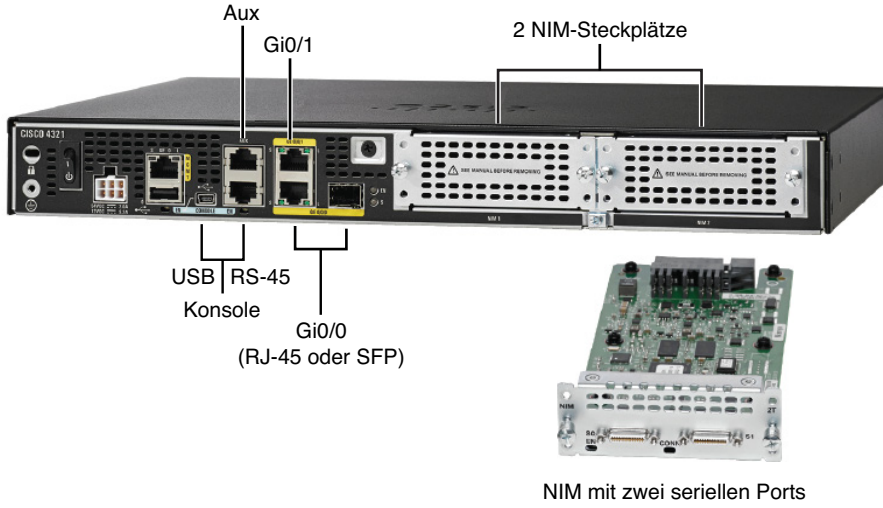


Abbildung P.6 Router mit Serial NIM auf der rechten Seite

Standleitungen und das T-Carrier-System

Die Provider bieten Standleitungen mit vielen unterschiedlichen Übertragungsraten an. Doch die Kunden des Serviceproviders können sich die Datenrate nicht einfach aussuchen. Diese richtet sich vielmehr nach einer recht alten Technologie: dem T-Carrier-System.

In den 50er- und 60er-Jahren des vorigen Jahrhunderts entwickelten die Unternehmen des Bell-Konzerns aus den USA die digitale Sprachverarbeitung und das T-Carrier-System. Im Rahmen dieser Entwicklung wurden dort die unterschiedlichen Übertragungsgeschwindigkeiten standardisiert, z. B. 64 Kbit/s, 1,544 Mbit/s und 44,736 Mbit/s.

Außerdem wurde dort die TDM-Technologie (Time-Division Multiplexing) entwickelt, durch die sich mehrere dieser Basisgeschwindigkeiten in einer einzigen Leitung kombinieren ließen. Beispielsweise ist Digital Signal Level 1 (DS1) oder T1 ein populärer Standard, der 24 DS0s (bei 64 Kbit/s) zuzüglich 8 Kbit/s Overhead in einer physischen Leitung kombiniert, die mit 1.544 Mbit/s betrieben wird. Um jedoch Kunden flexible Datenraten anbieten zu können, kann der Serviceprovider an vielen Einsatzorten eine T1-Leitung einbauen, einige davon aber mit geringerer und andere mit höherer Rate betreiben – solange es sich bei diesen Datenraten um Vielfache von 64 Kbit/s handelt.

Nun wieder zurück zur Theorie über die Geschwindigkeit einer Standleitung. Was findet sich also dafür für Sie im Angebot? Im Wesentlichen bekommen Sie bei langsameren Geschwindigkeiten verschiedene Vielfache von 64 Kbit/s bis hin zur T1-Geschwindigkeit. Bei schnelleren Geschwindigkeiten sind Vielfache der T1-Geschwindigkeit erhältlich – bis hin zur T3-Geschwindigkeit. In Tabelle P.2 werden die Geschwindigkeiten zusammengefasst.

Tabelle P.2 Übersicht über WAN-Datenraten

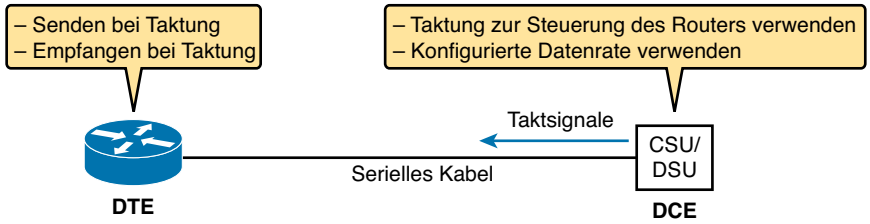
Schlüsselthema

Bezeichnung der Verbindung	Bitrate
DS0	64 Kbit/s
Fractional T1	Vielfache von 64 Kbit/s, bis hin zum 24-Fachen
DS1 (T1)	1.544 Mbit/s (24 DS0s, für 1.536 Mbit/s, plus 8 Kbit/s Overhead)
Fractional T3	Vielfache von 1.536 Kbit/s, bis hin zum 28-Fachen
DS3 (T3)	44.736 Mbit/s (28 DS1s, plus Management-Overhead)

Die Rolle der CSU/DSU

Für unsere Ausführungen über WAN-Verbindungen in einem Unternehmensnetzwerk wollen wir abschließend die Rolle des CSU/DSU (kurz CSU) betrachten. Wir gehen bei unseren Ausführungen in den folgenden Absätzen bis zu Abbildung P.7 von einer Standleitung mit externer CSU/DSU wie in Abbildung P.4 aus.

Die CSU befindet sich zwischen der Standleitung des Serviceproviders und dem Router und versteht sozusagen die Sprache beider Welten und deren Konventionen in Layer 1. Für den Serviceprovider bedeutet das, dass die CSU mit seiner Leitung verbunden ist und somit alle Details über das T-Carrier-System, TDM und die vom Serviceprovider eingesetzte Geschwindigkeit verstehen muss. Auf der anderen Seite ist die CSU mit dem Router verbunden, wobei die Rollen DCE bzw. DTE zum Einsatz kommen. Die CSU agiert als DCE (Data Circuit-Terminating Equipment) und steuert die Geschwindigkeit des seriellen Interface am Router. Der Router agiert als DTE (Data Terminal Equipment) und wird von den Taktungssignalen von der CSU (DCE) gesteuert. Das bedeutet, die CSU sagt dem Router, wann er Daten senden und empfangen soll; der Router versucht, die Bits nur dann zu senden und zu empfangen, wenn das DCE für die korrekten elektrischen Impulse (die sogenannte Taktung, auch Clocking genannt) in der Leitungsverbindung sorgt. Abbildung P.7 zeigt ein Diagramm dieser Grundkonzepte der Rolle der CSU/DSU.



Schlüssel-thema

Abbildung P.7 DCE- und DTE-Rollen einer CSU/DSU und eines seriellen Router-Interfaces

WAN-Verbindung im Lab einrichten

Vom Praktischen her sei erwähnt, dass Sie gut beraten sind, wenn Sie sich als Vorbereitung auf die CCENT- und CCNA R&S-Prüfungen gebrauchte Router- und Switch-Hardware zulegen, um damit praktische Erfahrungen zu sammeln. In diesem Fall können Sie eine Standleitung auch nachstellen, ohne die Dienste eines Providers in Anspruch nehmen oder eine CSU/DSU einsetzen zu müssen. Hierfür ist lediglich ein kleiner Trick erforderlich. Mit der folgenden Anleitung können Sie in Ihrem Labor zu Hause eine WAN-Verbindung einrichten.

Wenn eine echte WAN-Verbindung mit einem echten Serviceprovider zwischen zwei Standorten aufgebaut werden soll, werden die seriellen Kabel, die normalerweise für einen Router mit einer externen CSU/DSU verwendet werden, als *DTE-Kabel* bezeichnet. Folglich sind die seriellen Kabel in Abbildung P.4 weiter oben DTE-Kabel.

Sie können eine entsprechende WAN-Verbindung erstellen, indem Sie einfach zwei serielle Interfaces von Routern mit einem normalen DTE-Kabel und einem etwas anders gestalteten DCE-Kabel ohne CSUs und ohne Standleitung vom Serviceprovider verbinden. Das DCE-Kabel hat eine Buchse, das DCE-Kabel einen Stecker, d. h., die beiden Kabel können direkt miteinander verbunden werden. Damit wird die physische Verbindung geschlossen und die Daten können ihren Weg nehmen. Zudem erfüllt das DCE-Kabel die gleiche Funktion wie ein Ethernet-Crossover-Kabel, indem es die Sende- und Empfangsleiterpaare gegeneinander vertauscht (Abbildung P.8).

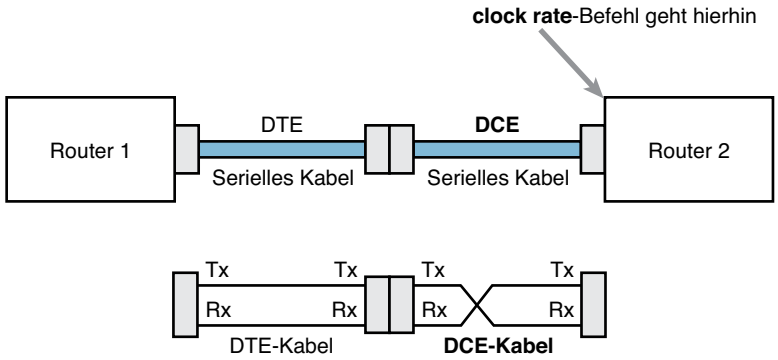


Abbildung P.8 Serielle Verkabelung mit DTE- und DCE-Kabel

Die Abbildung zeigt die Details oben für die Kabel und unten für die Leiter im Kabelinnern. Beachten Sie unten in der Abbildung vor allem, dass das serielle DTE-Kabel als Straight-Through-Kabel fungiert, weil hier – anders als beim DCE-Kabel – die Sende- und Empfangsleiterpaare nicht vertauscht sind.

HINWEIS Praktischerweise bieten viele Hersteller Einzelkabel an, in dem die beiden Kabel aus Abbildung P.8 kombiniert werden. Suchen Sie dazu online nach »Cisco serial crossover«.

Damit die Verbindung funktioniert, muss der Router mit dem angeschlossenen DCE-Kabel für die Taktung sorgen. Ein Router kann die Taktung zwar über ein serielles Interface bereitstellen, doch tut er dies nur, wenn an das Interface ein DCE-Kabel angeschlossen ist und er mit dem Befehl **clock rate** konfiguriert wurde. Neuere IOS-Versionen erkennen das Vorhandensein eines DCE-Kabels und legen automatisch eine Taktrate fest, die für ein Funktionieren der Verbindung sorgt. Unter älteren IOS-Versionen hingegen müssen Sie den Befehl **clock rate** konfigurieren.

Layer-2-Standleitungen mit HDLC

Standleitungen stellen Layer-1-Services bereit. Es wird also garantiert, dass Bits zwischen den an eine Standleitung angeschlossenen Geräten übertragen werden. Die Standleitung selbst definiert jedoch kein Data-Link-Layer-Protokoll, das für die Übertragung über eine Standleitung verwendet werden muss. HDLC enthält die Option eines Data-Link-Protokolls für eine Standleitung.

HDLC muss nur einige wenige große Aufgaben ausführen, weil eine Point-to-Point-Standleitung lediglich eine einfache Point-to-Point-Topologie hat. Zuerst entnimmt der empfangende Router dem Frame-Header, dass ein neuer Frame kommt. Außerdem hat der HDLC-Trailer – wie alle anderen Data-Link-Protokolle – ein FCS-Feld (Frame Check Sequence), das der empfangende Router nutzen kann, um zu entscheiden, ob der Frame fehlerhaft ist, und falls ja, diesen zu löschen.

Cisco ergänzt das ISO-Standard-HDLC-Protokoll um eine weitere Funktion, indem ein zusätzliches Feld (Typ) in den HDLC-Header eingefügt wird. So wird eine Cisco-spezifische Version von HDLC erstellt – siehe Abbildung P.9. Mit dem Type-Feld können Cisco-Router verschiedene Arten von Network-Layer-Paketen über die HDLC-Verbindung schicken. Zum Beispiel kann eine HDLC-Verbindung zwischen zwei Cisco-Routern sowohl IPv4- als auch IPv6-Pakete weiterleiten, weil sie anhand des Type-Felds identifizieren können, welcher Pakettyp im jeweiligen HDLC-Frame gekapselt ist.

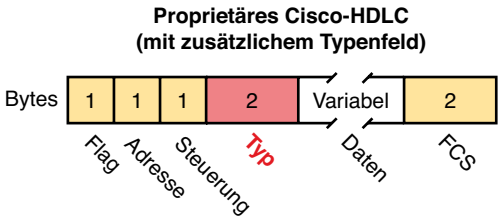


Abbildung P.9 Cisco-HDLC-Framing

Heute haben die HDLC-Adress- und Kontrollfelder kaum etwas zu tun. Wenn beispielsweise nur zwei Router miteinander verbunden sind, ist es klar, wenn einer von ihnen einen Frame sendet, dass der andere den bekommen soll. Früher waren die Address- und Control-Felder mal ziemlich wichtig, aber heute nicht mehr.

Router nutzen HDLC wie jedes andere Data-Link-Protokoll – nämlich dazu, Pakete zum nächsten Router zu transportieren. Abbildung P.10 zeigt drei bekannte Routing-Schritte, wobei sich die HDLC-Rolle in Schritt 2 befindet.

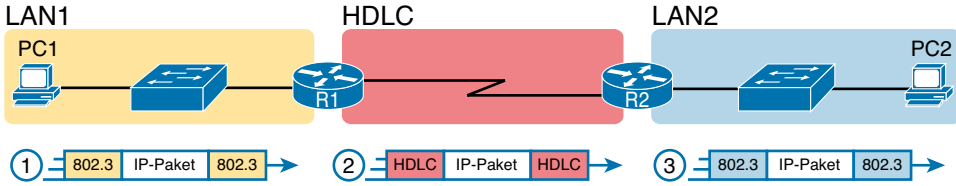


Abbildung P.10 Allgemeine Darstellung der Entkapselung und Neukapselung von IP-Paketen auf Routern

Gehen wir die einzelnen Schritte aus der Abbildung einmal durch:

1. Zum Versenden des IP-Pakets an Router R1 kapselt PC1 das IP-Paket in ein Ethernet-Frame.
2. Router R1 entkapselt das IP-Paket (d. h. entnimmt es aus dem Ethernet-Frame), kapselt es in einen HDLC-Frame mit HDLC-Header und -Trailer und leitet es an Router R2 weiter.
3. Router R2 entkapselt das IP-Paket, kapselt es in einen Ethernet-Frame und leitet es an PC2 weiter.

Zusammengefasst erstellt eine Standleitung mit HDLC eine WAN-Leitung zwischen zwei Routern, damit diese Router Pakete für die Geräte in den angeschlossenen LANs weiterleiten können. Die Standleitung selbst stellt das physische Mittel zur Übertragung der Bits in beide Richtungen bereit. Die HDLC-Frames sind das Mittel zur ordnungsgemäßen Kapselung des Network-Layer-Pakets, damit dieses über die Leitung zwischen den Routern übertragen werden kann.

HDLC konfigurieren

Erinnern Sie sich noch an die Ethernet-Interfaces bei Routern? Für diese Interfaces ist in Layer 1 und 2 keine Konfiguration nötig, damit das Interface funktioniert und IP-Datenverkehr weiterleitet. Die Details von Layer 1 erfolgen standardmäßig, sobald die Verkabelung korrekt installiert wurde. Die Ethernet-Interfaces der Router verwenden standardmäßig natürlich Ethernet als Data-Link-Protokoll. Der Router muss für das Interface nur eine IP-Adresse konfigurieren und es eventuell mit dem Befehl **no shutdown** aktivieren, falls es sich im Status *administratively down* befindet.

Entsprechend brauchen serielle Interfaces bei Cisco-Routern keine spezifischen Konfigurationsbefehle für Layer 1 oder 2. Für Layer 1 muss die Verkabelung natürlich abgeschlossen sein, aber der Router versucht, das serielle Interface zu nutzen, sobald der Befehl **no shutdown** konfiguriert wurde. Bei Layer 2 arbeitet das IOS bei seriellen Interfaces standardmäßig mit HDLC. Wie bei Ethernet-Interfaces brauchen auch serielle Router-Interfaces normalerweise nur den Befehl **ip address** und vielleicht noch **no shutdown**, sofern die Interfaces beider Router ansonsten die Default-Einstellungen aufweisen.

Konfigurations-Checkliste

Allerdings gibt es noch viele optionale Befehle für serielle Verbindungen. In der folgenden Liste werden einige Konfigurationsschritte skizziert und die Bedingungen aufgeführt, wann bestimmte Befehle nötig sind, plus einige, die rein optional sind:

Schritt 1: Konfigurieren Sie mit dem Befehl **ip address** *adresse maske* im Interfacekonfigurationsmodus die IP-Adresse des Interface.

Schritt 2: Die folgenden Schritte sind nur erforderlich, wenn die speziell aufgeführten Bedingungen zutreffen:

- **a.** Falls der Interfacesubbefehl **encapsulation** *Protokoll* für ein Nicht-HDLC-Protokoll existiert, aktivieren Sie HDLC mit dem Interfacesubbefehl **encapsulation hdlc**. Alternativ verwenden Sie den Befehl **no encapsulation** *protokoll* im Interfacekonfigurationsmodus, um auf die Verwendung von HPLC als Data-Link-Protokoll zurückzuschalten.
- **b.** Wenn das Interface den Line-Status *administratively down* hat, aktivieren Sie es mit dem Befehl **no shutdown** Interfacekonfigurationsmodus.
- **c.** Falls es sich bei der seriellen Verbindung um eine Back-to-Back-Leitung in einem Lab (oder einem Simulator) handelt, konfigurieren Sie die Taktrate mit dem Befehl **clock rate** *datenrate* im Interfacekonfigurationsmodus. Verwenden Sie diesen Befehl allerdings nur auf einem Router mit dem DCE-Kabel (siehe die Ausgabe des Befehls **show controllers serial** *nummer*).

Schritt 3: Die folgenden Schritte sind immer optional und wirken sich nicht darauf aus, ob die Verbindung funktioniert und IP-Datenverkehr durchlässt:

- **a.** Konfigurieren Sie die dokumentierte Datenrate der Verbindung mit dem Befehl **bandwidth** *datenrate-in-kbit/s* im Interfacekonfigurationsmodus so, dass diese der tatsächlichen Taktrate der Verbindung entspricht.
- **b.** Konfigurieren Sie zu Dokumentationszwecken mit dem Befehl **description** *text* eine Beschreibung zum Zweck des Interface.

Wenn Sie in der Praxis einen Cisco-Router ohne bereits vorhandene Interface-Konfiguration konfigurieren und eine normale produktive serielle Verbindung mit CSU/DSUs installieren, brauchen Sie wahrscheinlich nur die Konfigurationsbefehle **ip address** und **no shutdown**.

Abbildung P.11 zeigt ein Beispielnetzwerk und Listing P.1 die entsprechende HDLC-Konfiguration. In diesem Fall wurde die serielle Verbindung in einem Lab mit einer seriellen Back-to-Back-Verbindung erstellt, was die Schritte 1 (**ip address**) und 2C (**clock rate**) aus der vorigen Liste erfordert. Gezeigt wird auch der optionale Schritt 3b (**description**).

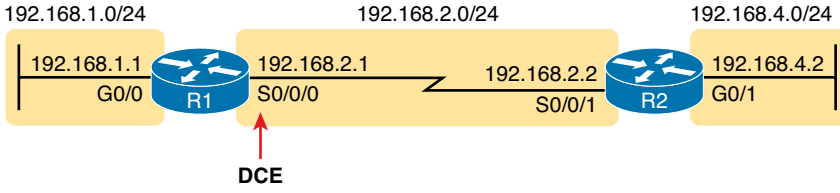


Abbildung P.11 Typische serielle Verbindung zwischen zwei Routern

Listing P.1 HDLC-Konfiguration

```
R1# show running-config
! Hinweis - es werden nur die relevanten Zeilen gezeigt
interface GigabitEthernet0/0
 ip address 192.168.1.1 255.255.255.0
!
interface Serial0/0/0
 ip address 192.168.2.1 255.255.255.0
 description link to R2
 clock rate 2000000
!
router eigrp 1
 network 192.168.1.0
 network 192.168.2.0
```

Die Konfiguration von R1 ist relativ einfach. Bei der entsprechenden Konfiguration des R2-Interface S0/0/1 braucht man nur den Befehl **ip address** plus die Default-Einstellungen **encapsulation hdlc** und **no shutdown.** Den Befehl **clock rate** braucht man bei R2 nicht, weil das DCE-Kabel in R1 steckt, also muss das DTE-Kabel bei R2 eingesteckt sein.

Listing P.2 führt zwei Befehle auf, die die Konfiguration von R1 und einige andere Default-Einstellungen bestätigen. Zuerst wird der Output des Befehls **show controllers** für S0/0/0 gezeigt, der bestätigt, dass tatsächlich ein DCE-Kabel in R1 steckt und dass die Taktrate auf 2.000.000 Bit/s gesetzt wurde. Der Befehl **show interfaces S0/0/0** listet oben die verschiedenen Konfigurationseinstellungen auf, z. B. den Default-Kapselungswert (HDLC) und die Default-Bandbreiteneinstellung für ein serielles Interface (1544, was für 1544 Kbit/s oder 1,544 Mbit/s steht). Außerdem finden sich hier die IP-Adresse, die Maske im Präfixstil (/24) sowie die Beschreibung wie in Listing P.1 konfiguriert.

Listing P.2 Die Konfigurationseinstellungen von R1 überprüfen

```
R1# show controllers serial 0/0/0
Interface Serial0/0/0
Hardware is SCC
DCE V.35, clock rate 2000000
! Zeilen aus Platzgründen gekürzt

R1# show interfaces s0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is WIC MBRD Serial
  Description: link to R2
  Internet address is 192.168.2.1/24
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:01, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
```

```
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     276 packets input, 19885 bytes, 0 no buffer
     Received 96 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     284 packets output, 19290 bytes, 0 underruns
     0 output errors, 0 collisions, 5 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
     7 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

Der Router verwendet das serielle Interface schließlich nur, wenn es sich im Interfacestatus *up/up* befindet, wie man der ersten Zeile des Outputs aus dem Befehl **show interfaces S0/0/0** in Listing P.2 entnehmen kann. Allgemein kann man sagen, dass sich das erste Statuswort auf den Status von Layer 1 bezieht und das zweite auf den Status von Layer 2. Um sich den Interfacestatus schneller anzuschauen, nehmen Sie stattdessen entweder den Befehl **show ip interface brief** oder **show interfaces description**, wie sie in Listing P.3 aufgeführt werden.

Listing P.3 Kurze Auflistung der Interfaces und ihr Status

```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0         192.168.1.1     YES manual up                    up
GigabitEthernet0/1         unassigned      YES manual administratively down down
Serial0/0/0                192.168.2.1     YES manual up                    up
Serial0/0/1                unassigned      YES NVRAM  administratively down down
Serial0/1/0                unassigned      YES NVRAM  administratively down down
Serial0/1/1                unassigned      YES NVRAM  administratively down down

R1# show interfaces description
Interface                      Status         Protocol Description
Gi0/0                          up             up       LAN at Site 1
Gi0/1                          admin down     down
Se0/0/0                        up             up       link to R2
Se0/0/1                        admin down     down
Se0/1/0                        admin down     down
Se0/1/1                        admin down     down
```

Standleitungs-WANs mit PPP

Das Point-to-Point Protocol (PPP) spielt die gleiche Rolle wie HDLC: ein Data-Link-Protokoll für serielle Verbindungen. Allerdings wurde HDLC in einer Welt ohne Router geschaffen. Im Gegensatz dazu wurde PPP in den 1990-er Jahren im Hinblick auf Router, TCP/IP und andere Network-Layer-Protokolle designt und weist viele sehr fortschrittliche Features auf.

Dieser zweite Hauptabschnitt beschäftigt sich zuerst mit den PPP-Konzepten und darunter beispielhaft auch eine fortschrittliche PPP-Eigenschaft (die Authentifizierung). Der Abschnitt endet mit einigen Konfigurationsbeispielen für PPP.

PPP-Konzepte

PPP enthält verschiedene einfache, aber wichtige Funktionen, die für Standleitungen zur Verbindung zweier Geräte praktisch sind:

Schlüsselthema

- Die Definition eines Headers und eines Trailers, die die Auslieferung eines Daten-Frames über die Verbindung ermöglichen
- Die Unterstützung synchroner und asynchroner Verbindungen
- Das Protokolltypfeld im Header, um mehrere Layer-3-Protokolle über dieselbe Verbindung übertragen zu können
- Die integrierten Authentifizierungstools: Password Authentication Protocol (PAP) und Challenge Handshake Authentication Protocol (CHAP)
- Die Steuerprotokolle für alle Protokolle übergeordneter Layer, die über PPP übertragen werden, zur einfacheren Integration und Unterstützung dieser Protokolle

Auf den nächsten Seiten werfen wir einen genaueren Blick auf das Protokollfeld, die Authentifizierung und die Steuerprotokolle.

PPP-Framing

Anders als die Standardversion von HDLC definiert der PPP-Standard ein Protokollfeld. Das Protokollfeld bezeichnet den Typ des im Frame gekapselten Pakets. Bei der Einführung von PPP war dieses Feld dafür vorgesehen, Pakete vieler verschiedener Layer-3-Protokolle über dieselbe Verbindung zu übertragen. Auch heute bietet das Protokollfeld diese Funktionalität und unterstützt dabei sogar Pakete für verschiedene IP-Versionen (IPv4 und IPv6). Abbildung P.12 zeigt das PPP-Framing, das dem Cisco-proprietären HDLC-Framing entspricht und ein Protokolltypfeld enthält (wie bereits in Abbildung P.9 gezeigt).

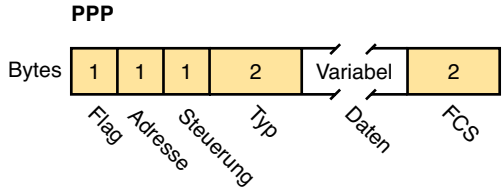


Abbildung P.12 PPP-Framing

PPP-Steuerprotokolle

Neben dem Framing wie bei HDLC definiert PPP eine Gruppe von Layer-2-Steuerprotokollen, die verschiedene Verbindungssteuerungsfunktionen ausführen. Diese zusätzlichen Protokolle funktionieren im Prinzip auf ähnliche Weise, wie Ethernet zusätzliche Protokolle wie STP (Spanning Tree Protocol) einbindet. Ethernet verfügt über Header und Trailer, um Frames zu übermitteln, und definiert außerdem Overhead-Protokolle wie STP, damit der Prozess der Frame-Weiterleitung reibungsloser funktioniert. Entsprechend definiert PPP das Frame-Format in Abbildung P.12 und zusätzlich andere Protokolle, um die serielle Verbindung besser verwalten und steuern zu können.

PPP teilt diese Steuerprotokolle in zwei separate Hauptkategorien auf:

Schlüsselthema

- **Link Control Protocol (LCP):** Dieses Protokoll verfügt über mehrere verschiedene Funktionen, die sich jeweils auf den Data Link Layer selbst konzentrieren und das über die Verbindung gesendete Layer-3-Protokoll ignorieren.
- **Network Control Protocols (NCP):** Dies ist eine Kategorie von Protokollen, eines pro Network-Layer-Protokoll. Jedes Protokoll kümmert sich um Funktionen, die für das dazugehörige Layer-3-Protokoll spezifisch sind.

Das PPP-LCP implementiert die Steuerfunktionen, die unabhängig vom Layer-3-Protokoll immer gleich funktionieren. Für Features, die mit einem Protokoll eines höheren Layer zusammenhängen (meist Layer-3-Protokolle), verwendet PPP eine Reihe von PPP *Control Protocols* (CP), z. B. das IP Control Protocol (IPCP). PPP arbeitet mit einer Instanz von LCP pro Verbindung und einem NCP für jedes in der Verbindung definierte Layer-3-Protokoll. Wenn beispielsweise in einer PPP-Verbindung IPv4, IPv6 und das Cisco Discovery Protocol (CDP) verwendet werden, nutzt die Verbindung eine Instanz von LCP plus IPCP (für IPv4), IPv6CP (für IPv6) sowie CDPCP (für CDP).

In Tabelle P.3 werden die Funktionen von LCP zusammengefasst, die Namen der LCP-Eigenschaften aufgeführt und die Features kurz beschrieben. Im Anschluss an die Tabelle wird eines der Features, die PPP-Authentifizierung, ausführlicher behandelt. Weiter unten im Abschnitt »MLPPP konfigurieren« werden wir MLPPP (Multilink PPP) behandeln.

Tabelle P.3 PPP LCP-Merkmale

Funktion	LCP-Merkmal	Beschreibung
Erkennung von Loops auf der Verbindung	Magic Number	Erkennt, ob ein Loop auf der Verbindung vorliegt, und deaktiviert ggf. das Interface. Das Routing erfolgt dann über eine funktionsfähige Route.
Fehlererkennung	Link-Quality Monitoring (LQM)	Deaktiviert ein Interface, bei dem ein bestimmter Fehlerschwellwert überschritten wird. Das Routing erfolgt dann über bessere Routen.
Unterstützung mehrerer Verbindungen	Multilink-PPP	Führt einen Lastausgleich (Load Balancing) auf mehreren parallelen Verbindungen durch.
Authentifizierung	PAP und CHAP	Gestattet den Austausch von Namen und Passwörtern, sodass jedes Gerät die Identität des Geräts am anderen Ende der Verbindung verifizieren kann.

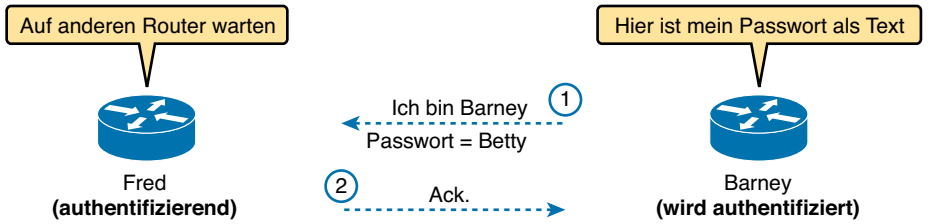
PPP-Authentifizierung

Beim Networking bekommt ein Gerät durch die *Authentifizierung* die Möglichkeit zu bestätigen, dass ein anderes Gerät wirklich das korrekte und anerkannte Gerät ist, mit dem die Kommunikation stattfinden soll. Anders gesagt wird durch die Authentifizierung bestätigt, dass die Gegenseite wirklich der authentische andere Teilnehmer ist und kein Betrüger.

Wenn beispielsweise R1 und R2 mit PPP über eine serielle Verbindung kommunizieren sollen, verlangt R1 von R2 unter Umständen einen Nachweis darüber, dass es sich bei R2 tatsächlich um R2 handelt. In diesem Szenario will R1 von R2 die Authentifizierung und R2 kann über den Authentifizierungsprozess R1 gegenüber seine Identität beweisen.

Die WAN-Authentifizierung wird meistens bei Wählverbindungen eingesetzt. Allerdings bleibt unabhängig davon, ob eine Wähl- oder Standleitung verwendet wird, die Konfiguration der Authentifizierungsmerkmale stets gleich.

PPP definiert zwei Authentifizierungsprotokolle: PAP und CHAP. Beide Protokolle erfordern den Austausch von Nachrichten zwischen Geräten, aber mit unterschiedlichen Details. Bei PAP beginnt der Vorgang mit dem Gerät, das sich authentifizieren soll, welches dann die Nachrichten sendet und seine Legitimation darlegt, indem ein geheimes Passwort im Klartext aufgelistet wird (siehe Abbildung P.13).



Schlüssel-thema

Abbildung P.13 PAP-Authentifizierungsprozess

Wenn wie in der Abbildung die Verbindung zustande kommt, erfolgt die Authentifizierung in zwei Schritten. In Schritt 1 sendet Barney das gemeinsame Passwort im Klartext. Fred, der von Barney eine Authentifizierung braucht (also bestätigt wissen will, dass Barney tatsächlich der echte Barney ist), betrachtet das Passwort. Weil Barneys Name und Passwort auf Fred konfiguriert sind, überprüft Fred diese Konfiguration, stellt fest, dass das Passwort korrekt ist, und schickt seinerseits eine Bestätigung zurück, aus der hervorgeht, dass Barney die Authentifizierung bestanden hat.

CHAP ist eine weitaus sicherere Option und verwendet verschiedene Nachrichten, wobei das Passwort verborgen bleibt. Bei CHAP beginnt das Gerät, das die Authentifizierung vornimmt (Fred), mit einer Nachricht, die man als *Challenge* bezeichnet und die das andere Gerät zum Antworten auffordert. Der große Unterschied besteht darin, dass im Ablauf die zweite Nachricht (siehe Abbildung P.14) das Authentifizierungspasswort versteckt, indem stattdessen eine gehashte Version des Passworts gesendet wird. Auf dem Router Fred wurden Name und Passwort von Barney so vorkonfiguriert, dass Fred überprüfen kann, ob der von Barney gesendete Passwort-Hash tatsächlich zu dem Passwort gehört, das in Freds Konfiguration für Barney aufgelistet ist. Sofern das Passwort tatsächlich korrekt ist, sendet Fred eine dritte Nachricht zurück, um die erfolgreiche Authentifizierung von Barney zu bestätigen.

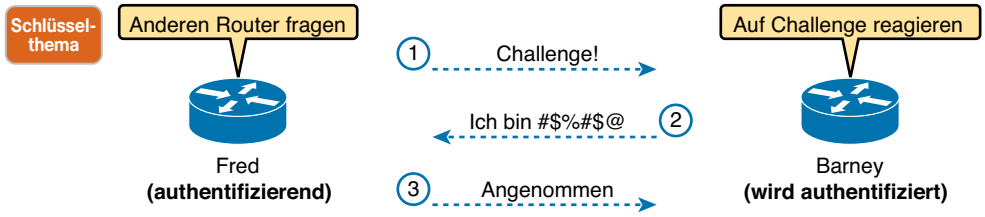


Abbildung P.14 CHAP-Authentifizierungsprozess

Die Abbildungen P.13 und P.14 zeigen die Abläufe bei funktionierender Authentifizierung. Läuft etwas schief (wenn z. B. die Passwörter nicht übereinstimmen), wird am Schluss eine andere Nachricht übertragen. Wenn die Authentifizierung misslingt, belässt PPP das Interface im Status *up/down* und der Router kann keine Frames über dieses Interface weiterleiten und empfangen.

Die Datenübertragung ist bei PAP wesentlich unsicherer als bei CHAP, da PAP den Hostnamen und das Passwort unverschlüsselt sendet. Diese Daten lassen sich mithilfe eines Nachverfolgungstools, das in die Leitung eingeschleift wird, von Dritten relativ einfach auslesen. CHAP verwendet stattdessen einen unidirektionalen Hash-Algorithmus namens MD5 (Message Digest 5). Der Eingabewert dieses Algorithmus ist ein Passwort, das selbst zu keinem Zeitpunkt übertragen wird, sowie eine gemeinsam verwendete Zufallszahl.

Der CHAP-Prozess verwendet außerdem diesen Hash-Wert nur ein einziges Mal, sodass ein Angreifer nicht einfach den gehashten Wert kopieren und später senden kann. Damit das funktioniert, enthält die CHAP-Challenge (die erste CHAP-Nachricht) eine Zufallszahl. Der Ziel-Router führt dann den Hash-Algorithmus unter Verwendung der empfangenen Zufallszahl und des geheimen Passworts aus und sendet das Ergebnis an den anfragenden Router zurück. Dieser führt währenddessen denselben Algorithmus für die (zuvor übermittelte) Zufallszahl und das (nicht übermittelte, sondern lokal gespeicherte) Passwort aus. Stimmen beide Ergebnisse überein, müssen auch die Passwörter gleich sein. Wenn dann später erneut eine Authentifizierung vorgenommen werden soll, generiert und nutzt der authentifizierende Router eine andere Zufallszahl.

PAP und CHAP sind einige Beispiele der Arbeit, die das LCP von PPP erledigt. Das nächste Thema ist, wie man PPP konfiguriert und überprüft.

PPP implementieren

Für die Konfiguration von PPP muss man im Vergleich zu HDLC nur eines ändern: Man verwendet an beiden Enden der Verbindung den Befehl **encapsulation ppp**. Wie bei HDLC können auch andere Aspekte wie z. B. die Interface-**Bandbreite** und -**Beschreibung** optional konfiguriert werden. Und natürlich muss das Interface auch aktiviert werden (**no shutdown**). Aber um von HDLC zu PPP zu migrieren, braucht man die seriellen Interfaces beider Router nur mit dem Befehl **encapsulation ppp** zu konfigurieren.

Listing P.4 zeigt eine einfache Konfiguration unter Verwendung der beiden in Abbildung P.11 gezeigten Router, also dem gleichen Netzwerkverbund wie beim HDLC-Beispiel. Das Listing zeigt auch die Konfiguration der IP-Adressen, die allerdings für die Funktionsfähigkeit von PPP nicht konfiguriert sein müssen.

Listing P.4 Grundlegende PPP-Konfiguration

```
! Das Beispiel beginnt mit Router R1
interface Serial0/0/0
 ip address 192.168.2.1 255.255.255.0
 encapsulation ppp
 clockrate 2000000
! Als Nächstes die Konfiguration auf Router R2
interface Serial0/0/1
 ip address 192.168.2.2 255.255.255.0
 encapsulation ppp
```

Der **show**-Befehl zum Auflisten der PPP-Details lautet **show interfaces**, hier mit einem Beispiel von R1 aus Listing P.5. Die Ausgabe sieht bis zur ersten, grau hinterlegten Zeile im Listing genauso aus wie für HDLC. Die beiden hervorgehobenen Zeilen bestätigen die Konfiguration (»encapsulation PPP«). Diese Zeilen verweisen außerdem darauf, dass LCP seine Arbeit erfolgreich abgeschlossen hat (erkennbar an der Formulierung »LCP Open«). Schließlich können wir der Ausgabe noch entnehmen, dass zwei CPs (CDPCP und IPCP) ebenfalls erfolgreich aktiviert wurden – alles verlässliche Anzeichen dafür, dass PPP tatsächlich korrekt funktioniert.

Listing P.5 Den Status von PPP, LCP und NCP über **show interfaces** herausfinden

```
R1# show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is WIC MBRD Serial
  Description: link to R2
  Internet address is 192.168.2.1/24
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: IPCP, CDPCP, loopback not set
! Zeilen aus Platzgründen gekürzt
```

PPP CHAP implementieren

Die einfachste Version der CHAP-Konfiguration erfordert nur ein paar Befehle. Die Konfiguration verwendet ein Passwort, das auf jedem Router konfiguriert ist. (Als Alternative könnte das Passwort auch außerhalb des Routers auf einem AAA-Server (Authentication, Authorization, Accounting) konfiguriert werden.)

Konfigurations-Checkliste

Zur Konfiguration von PPP mit CHAP auf einem Interface, für das alle Default-Einstellungen für serielle Interfaces auf beiden Routern gelten, gehen Sie wie folgt vor:

Schritt 1: Setzen Sie für die seriellen Interfaces auf beiden Routern den Befehl **encapsulation ppp** Interfacekonfigurationsmodus ab, um PPP auf den Interfaces zu aktivieren.

Schritt 2: Definieren Sie die von den beiden Routern verwendeten Benutzernamen und Passwörter:

a. Setzen Sie auf beiden Routern jeweils den Befehl **hostname** *name* im globalen Konfigurationsmodus ab, um den Namen festzulegen, den der lokale Router bei der Authentifizierung verwenden wird.

b. Setzen Sie auf beiden Routern jeweils den Befehl **username** *name* **password** *passwort* im globalen Konfigurationsmodus ab, um den vom benachbarten Router verwendeten Namen und das zugehörige Passwort zu definieren. Beachten Sie, dass für beide Angaben die Groß-/Kleinschreibung unterschieden wird. (Der Name im Befehl **username** sollte mit dem im Befehl **hostname** auf dem benachbarten Router angegebenen übereinstimmen.)

Schritt 3: Geben Sie auf jedem Router den Befehl **ppp authentication chap** im Interfacekonfigurationsmodus ein, um CHAP für die betreffenden Interfaces zu aktivieren.

Abbildung P.15 zeigt die Konfiguration für R1 und R2, um bei der Verbindung PPP zu aktivieren und CHAP hinzuzufügen. Die Abbildung zeigt, wie der Name im Befehl **hostname** auf dem einen Router zum Befehl **username** auf dem anderen Router passen muss. Außerdem ist zu sehen, dass das in jedem **username**-Befehl definierte Passwort gleich sein muss (in diesem Fall mypass).

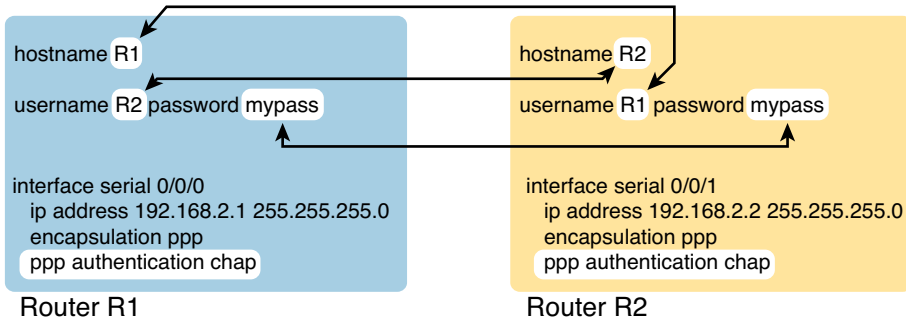


Abbildung P.15 CHAP-Konfiguration

Es gibt mehrere Möglichkeiten zu überprüfen, ob die CHAP-Authentifizierung erfolgreich war. Zunächst wird, wenn eine aktivierte CHAP-Authentifizierung fehlschlägt, der Protokollstatus des Interface auf den Status *down* umgestellt. Zum Überprüfen dieses Status verwenden Sie wie üblich einen der Befehle **show interfaces** [*typ nummer*] oder **show interfaces status.** Ferner zeigt, wenn die aktivierte Chat-Authentifizierung fehlschlägt, der Befehl **show interfaces** anders als in diesem Beispiel nicht »LCP Open« an. Listing P.6 zeigt die Ausgabe des Befehls **show interfaces serial0/0/0** auf R1. Dabei ist CHAP wie in Abbildung P.15 gezeigt aktiviert und funktioniert auch einwandfrei. Allerdings geht aus dem Befehl nicht hervor, ob die Authentifizierung tatsächlich konfiguriert wurde.

Listing P.6 CHAP-Authentifizierung mit **show interfaces** überprüfen

```
R1# show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is WIC MBRD Serial
  Description: link to R2
  Internet address is 192.168.2.1/24
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: IPCP, CDPCP, loopback not set
  Keepalive set (10 sec)
! Zeilen aus Platzgründen gekürzt
R1# show ppp all
Interface/ID OPEN+ Nego* Fail-     Stage    Peer Address    Peer Name
------------ --------------------- -------- --------------- --------------------
Se0/0/0      LCP+ CHAP+ IPCP+ CDP> LocalT   192.168.2.2     R2
```

Die näherliegende Möglichkeit, sich zu vergewissern, dass CHAP funktioniert, ist der Befehl **show ppp all**, wie wir ihn am Ende von Listing P.6 sehen. mit diesem Befehl wird für jede PPP-Verbindung auf dem Router eine Zeile ausgegeben. Der im Listing hervorgehobene Header ist die Spalte, in der dieser Befehl verschiedene PPP-Protokolle und ihren Status auflistet. Dabei bedeutet das Pluszeichen (+), dass das angegebene Protokoll offen ist, während das Minuszeichen (–) auf einen Protokollfehler hindeutet. Die im Listing hervorgehobenen Zeilen bestätigen, dass Serial 0/0/0 PPP mit der CHAP-Authentifizierung verwendet und dass die CHAP-Authentifizierung funktioniert hat (dies zeigte der Status *open* für das CHAP-Protokoll).

PPP PAP implementieren

Die PAP-Konfiguration unterscheidet sich in verschiedener Hinsicht von der CHAP-Konfiguration. Zunächst einmal verwendet PAP den sehr ähnlichen Befehl **authentication ppp pap** (anstelle von **authentication ppp chap**). Zudem konfiguriert PAP die übermittelte Benutzername-Passwort-Kombination ganz anders als CHAP. Ein Router definiert die Benutzername-Passwort-Kombination, die er zu versenden gedenkt, mit dem Befehl **ppp pap sent-username**, der als Interfacesubbefehl konfiguriert wird. Der Router auf der Gegenseite empfängt diese Kombination und vergleicht die Werte mit den verschiedenen globalen **username password**-Befehlen. Abbildung P.16 zeigt eine vollständige Konfiguration für zwei Router (R1 und R2), wobei der Schwerpunkt auf dem Vergleich des Befehls **ppp pap sent-username** auf dem einen Router mit den **username password**-Befehlen auf dem anderen Router liegt.

Listing P.7 zeigt zwei Befehle, die zur Überprüfung des PAP-Betriebs verwendet werden. Beachten Sie besonders, dass aus der Ausgabe von **show interfaces** nicht mehr und nicht weniger hervorgeht als das, was wir auch bei Verwendung der CHAP-Authentifizierung erfahren. Der Line-Status *up* bestätigt, dass die Authentifizierung – sofern konfiguriert – funktioniert hat. (Allerdings können wir der Ausgabe von **show interfaces** auch nicht entnehmen, ob CHAP oder PAP konfiguriert wurde.) Wie bei CHAP bestätigt der LCP-Status *open* auch hier, dass die Authentifizierung funktioniert hat (wobei wir natürlich davon ausgehen müssen, dass diese auch konfiguriert ist). Allerdings zeigt uns der Befehl wiederum nicht, ob CHAP, PAP oder überhaupt eine Authentifizierung konfiguriert wurde. Eine bessere Bestätigung finden wir in

Schlüssel-thema

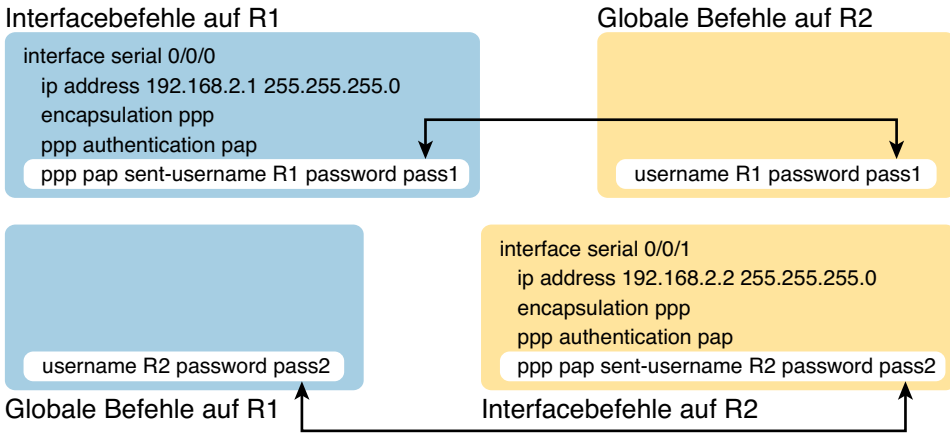


Abbildung P.16 PAP-Konfiguration

dem Befehl **show ppp all** ganz unten im Listing. Aus der Ausgabe geht hervor, dass PAP für das Interface Serial 0/0/0 konfiguriert wurde und den Protokollstatus *open* hat, was bedeutet, dass die Authentifizierung funktioniert hat.

Listing P.7 PAP-Authentifizierung konfigurieren und überprüfen

```
R1# show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is WIC MBRD Serial
  Description: link to R2
  Internet address is 192.168.2.1/24
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: IPCP, CDPCP, loopback not set
  Keepalive set (10 sec)
! Zeilen aus Platzgründen gekürzt
R1# show ppp all
Interface/ID OPEN+ Nego* Fail-     Stage    Peer Address    Peer Name
------------ -------------------- -------- --------------- --------------------
Se0/0/0      LCP+ PAP+ IPCP+ CDPC> LocalT   192.168.2.2     ciscouser2
```

MLPPP implementieren

Netzdesigner verwenden manchmal anstelle nur einer einzigen seriellen Leitung mehrere parallel angeordnete serielle Links zwischen zwei Routern. Dies kann aus Gründen einer besseren Verfügbarkeit wünschenswert sein: Wenn ein Link ausfällt, sind (hoffentlich) noch weitere funktionsfähige vorhanden. Auch die Wirtschaftlichkeit kommt als Motivation infrage. Schließlich könnte es preiswerter sein, zwei oder drei parallele T1-Leitungen mit je 1,5 Mbit/s zu installieren, statt sich für die nächstschnellere Leitungsform – eine T3-Leitung mit einem Fractional T3-Dienst – zu entscheiden. Wie auch immer: Am Ende steht ein Design mit mehreren seriellen Links zwischen zwei Routern, das so aussieht, wie in Abbildung P. 17 dargestellt.

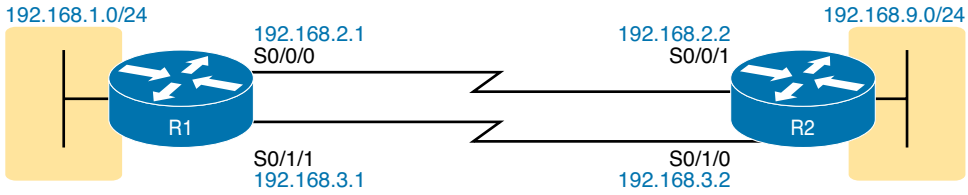


Abbildung P.17 Mehrere parallel angeordnete serielle Links zwischen Routern

Konfiguriert der Netzwerktechniker die parallel angeordneten seriellen Links wie gerade beschrieben, dann verfügt jeder Link über IP-Adressen und kann zur Weiterleitung von IP-Paketen verwendet werden. Zu diesem Zweck würde das Interior-Routing-Protokoll auf jedem der parallelen Links ausgeführt und die entsprechenden Nachbarschaftsbeziehungen würden über jeden Link gebildet werden. Infolgedessen würde auch jeder beteiligte Router mehrere Routen in jedes entfernte Zielnetzwerk erlernen (nämlich eine Route für jede parallele Leitung).

Abbildung P.18 zeigt das Konzept mehrerer gleichwertiger Routen mit je einer Route je parallel angeordneter serieller Verbindung. Wir sehen hier dasselbe Design wie in Abbildung P.17, mit zwei Links. R1 verfügt über je eine Route in das Netzwerk 192.168.9.0/24 über die obere und die untere Leitung. Bei Verwendung von EIGRP (Enhanced Interior Gateway Routing Protocol) hätte R1 zwei Nachbarschaftsbeziehungen mit R2, nämlich eine über jede Leitung.

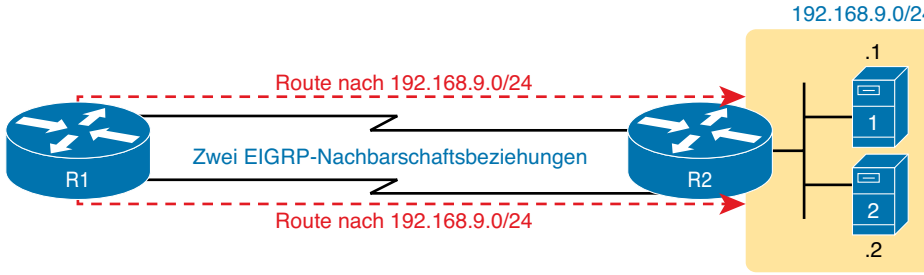


Abbildung P.18 Zwei IP-Routen in dasselbe Netzwerk – eine je parallel angeordneter serieller Leitung

Die Layer-3-Routing-Logik im Cisco IOS führt dann eine ausgewogene Verteilung der Pakete über die verschiedenen Links unter Verwendung der in der Abbildung gezeigten Routen. Normalerweise erfolgt diese Übertragungssymmetrierung auf Grundlage der jeweiligen Zieladresse. In Abbildung P.18 beispielsweise könnten alle Pakete an die Empfängeradresse 192.168.9.1 über die obere Leitung und alle Pakete an die Adresse 192.168.9.2 über die untere Leitung geroutet werden. Das IOS kann auch so konfiguriert werden, dass die Symmetrierung paketweise erfolgt.

Die soeben beschriebenen Layer-3-Funktionen funktionieren und in vielen Fällen tun sie das auch gut. Allerdings bietet PPP eine Funktion an, die den Layer-3-Betrieb in Topologien mit vielen parallelen PPP-Links ganz erheblich vereinfacht. Die Rede ist von Multilink PPP.

Multilink PPP-Konzepte

Multilink PPP (MLPPP) ist eine PPP-Funktion, die sich bei paralleler Verwendung vieler serieller Leitungen zwischen zwei Geräten als sehr praktisch erwiesen hat. MLPPP weist zwei wichtige Eigenschaften auf: Erstens reduziert es die Layer-3-Komplexität, indem es dafür sorgt, dass die vielen seriellen Interfaces auf den beteiligten Routern aus Layer-3-Perspektive wie ein einziges

Interface agieren. Statt also mehrere Subnetze mit Nachbarschaftsbeziehungen unterschiedlicher Routingprotokolle und mehreren gleichwertigen Routen zu betreiben, die für jedes Remote-Subnetz erlernt werden, benötigen Router für den Weg zum benachbarten Router nur noch ein einziges Subnetz, eine Nachbarschaftsbeziehung und eine Route je Zielsubnetz. Abbildung P.19 zeigt diese wesentlichen Konzepte für die physische Topologie aus Abbildung P.18.

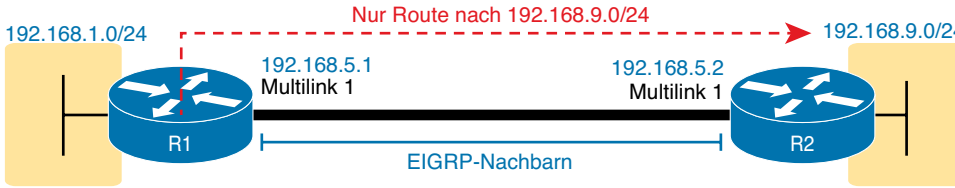


Abbildung P.19 Mit einem Multilink-Interface erstelltes Layer-3-Konzept

MLPPP sorgt dafür, dass mehrere physische Leitungen wie ein einzelner Link wirken. Hierzu wird ein virtuelles Interface verwendet: das sogenannte Multilink-Interface. Die Layer-3-Konfiguration (also IPv4- und IPv6-Adressen sowie Interfacesubbefehle des Routing-Protokolls) wird zum Multilink-Interface hinzugefügt. Danach verknüpft die Konfiguration die physischen seriellen Interfaces mit dem Multilink-Interface und verbindet dabei die Layer-2-Logik für diese mehreren Leitungen mit der Layer-3-Logik für das Multilink-Interface.

Hierdurch werden nicht nur wie bereits erwähnt die Layer-3-Details vereinfacht, sondern MLPPP führt auch einen Lastenausgleich für Frames durch, die auf Layer 2 über mehrere Links gesendet wurden. Bei MLPPP routet die Layer-3-Weiterleitungslogik jedes Paket über das Multilink-Interface. Wenn das IOS ein Paket intern über ein Multilink-Interface weiterleitet, übernimmt die Lastenausgleichslogik von MLPPP die Kontrolle, kapselt das Paket in einen neuen Data-Link-Frame und führt für diese Frames den Lastenausgleich durch.

Interessanterweise erfolgt der Lastenausgleich auf Frame-Ebene bei MLPPP dadurch, dass Frames in viele kleinere Frames unterteilt werden, die dann ihrerseits über jeweils einen Link übertragen werden. Abbildung P.20 veranschaulicht diesen Prozess. Die Schritte 1 und 2 zeigen das normale Routing, bei dem ein gekapseltes IP-Paket in Schritt 1 empfangen wird und der Router in Schritt 2 wie üblich eine Routing-Entscheidung trifft. Da allerdings das Paket über ein Multilink-Interface weitergeleitet wird, unterteilt MLPPP es in sogenannte Fragmente, die jeweils einen PPP-Header und -Trailer aufweisen. Einige wenige zusätzliche Header-Bytes sind dabei für die Verwaltung des Fragmentierungsvorgangs zuständig. Der empfangende Router setzt die Fragmente dann wieder zum Ursprungspaket zusammen (Schritt 4), woraufhin das normale IP-Routing erfolgt (Schritt 5).

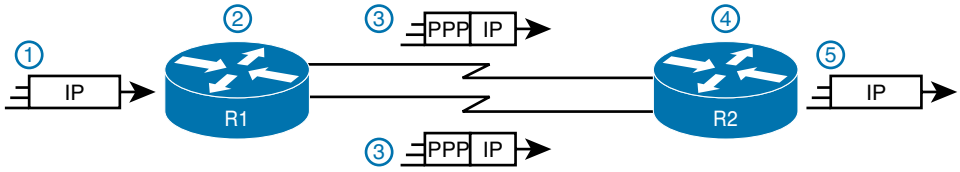


Abbildung P.20 Layer-2-Fragmentierung für den Lastenausgleich über mehrere Links

Der Lastenausgleichsprozess von MLPPP ermöglicht zwar geringfügige Schwankungen bei den Fragmentgrößen, doch in der Regel sorgen die Cisco-Router dafür, dass die Bytes gleichmäßig über die aktiven Links im Multilink-Bündel übertragen werden. Wenn also beispielsweise drei

Links aktiv sind, dann leitet der Router jeweils etwa ein Drittel des Traffic über jede Leitung weiter.

MLPPP konfigurieren

Die Implementierung von MLPPP erfordert eine etwas aufwendigere Konfiguration als die meisten anderen in diesem Buch behandelten Funktionen. Deswegen wollen wir, um einen Kontext zu schaffen, zunächst die folgenden drei wesentlichen Konfigurationsanforderungen für MLPPP erörtern:

Schlüsselthema

Schritt 1: Konfigurieren Sie die zueinander passenden Multilink-Interfaces auf den beiden beteiligten Routern. Konfigurieren Sie dabei alle Layer-3-Merkmale – IPv4, IPv6 und Routing-Protokoll – mit Interfacesubbefehlen für die Multilink-Interfaces (nicht für die seriellen Interfaces).

Schritt 2: Konfigurieren Sie für die seriellen Interfaces alle Layer-1- und Layer-2-Befehle, beispielsweise **clock rate** (Layer 1) oder die PPP-Authentifizierung (Layer 2).

Schritt 3: Konfigurieren Sie einige PPP-Befehle auf dem Multilink-Interface und den seriellen Interfaces, um MLPPP zu aktivieren und das Multilink-Interface mit den seriellen Interfaces zu verknüpfen.

Abbildung P.21 zeigt die spezifischen MLPPP-Befehle in einem Praxisbeispiel. Dieses basiert auf dem Design der Abbildungen P.19 und P.20. Beachten Sie, dass Abbildung P.21 aus Platzgründen nur die Konfiguration für eines der beiden seriellen Interfaces anzeigt; allerdings sind die Subbefehle für die seriellen Interfaces bei Verwendung von MLPPP ohnehin identisch.

Schlüsselthema

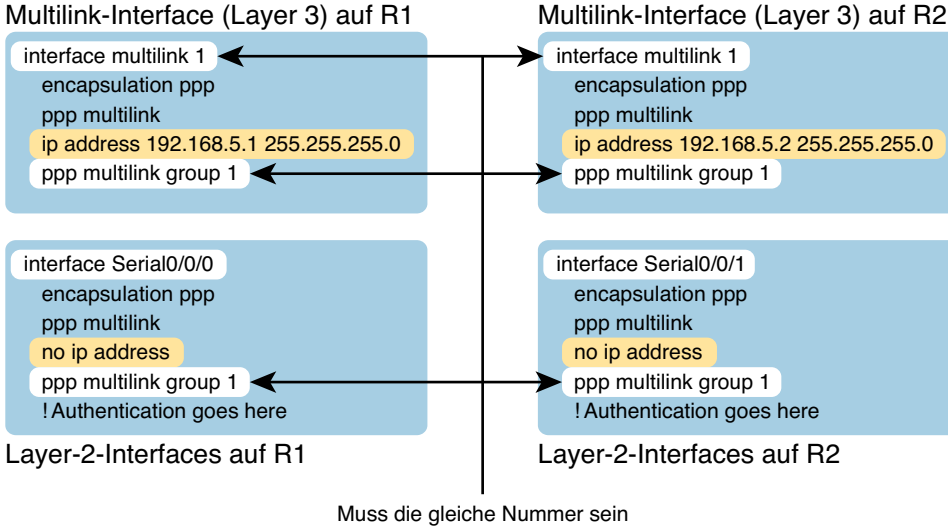


Abbildung P.21 MLPPP-Konfiguration

Betrachten wir zunächst die sechs Konfigurationsbefehle, die in Abbildung P.21 weiß unterlegt und durch die Pfeile gekennzeichnet sind. Der Befehl **interface multilink 1** erstellt auf jedem Router das Multilink-Interface. Der Netzwerktechniker wählt die Interfacenummer aus, doch muss diese auf beiden Routern nicht gleich sein, da sie nur lokale Bedeutung hat. Außerdem

muss für die Multilink-Interfaces wie auch für die physischen seriellen Interfaces der Befehl **ppp multilink group 1** angegeben sein, wobei auch die Nummer (in diesem Beispiel 1) auf allen Interfaces identisch sein muss. Sie können jede beliebige Nummer aus dem Bereich verwenden, nur muss diese bei den in der Abbildung hervorgehobenen Befehlen immer gleich sein.

Ebenfalls in der Abbildung hervorgehoben – wenn auch mit einer etwas dunkleren Farbe – sind die **ip address**-Befehle. Beachten Sie, dass in der Konfiguration IPv4-Adressen für die Multilink-Interfaces erscheinen, jedoch keine IPv4-Adresse für das serielle Interface. Kurz gesagt gibt es für das Multilink-Interface eine Layer-3-Konfiguration, für die seriellen Interfaces hingegen nicht. Infolgedessen nutzen Routing und Routing-Protokolllogik nur das Multilink-Interface.

Abschließend sei darauf hingewiesen, dass für das Multilink-Interface und für die seriellen Interfaces zwei weitere Befehle vorhanden sind: **encapsulation ppp** zur Aktivierung von PPP und **ppp multilink** zum Hinzufügen der Multilink-Unterstützung.

HINWEIS Abbildung P.21 zeigt nur ein serielles Interface, aber jedes serielle Interface in der Multilink-Gruppe weist ohnehin dieselbe Konfiguration auf.

MLPPP überprüfen

Wenn Sie überprüfen möchten, ob ein MLPPP-Interface einwandfrei funktioniert, sollten Sie sich die Layer-3-Funktionen separat von den Details für die Layer 1 und 2 vorstellen. Bei Layer 3 werden nun alle gängigen IPv4-, IPv6- und Routing-Protokollbefehle statt der physischen seriellen Interfaces das Multilink-Interface auflisten. Sie können auch einfach einen **ping**-Befehl an die IP-Adresse am anderen Ende des Multilinks senden, um diesen zu testen. Listing P.8 zeigt ein paar Befehle, mit denen der aktuelle Status des MLPPP-Links überprüft werden kann. Es wurde mit der funktionsfähigen Konfiguration aus Abbildung P.21 erstellt.

Listing P.8 Layer-3-Betrieb mit einem MLPPP-Multilink-Interface überprüfen

```
R1# show ip route
! Legende aus Platzgründen weggelassen

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/0
L        192.168.1.1/32 is directly connected, GigabitEthernet0/0
      192.168.5.0/24 is variably subnetted, 3 subnets, 2 masks
C        192.168.5.0/24 is directly connected, Multilink1
L        192.168.5.1/32 is directly connected, Multilink1
C        192.168.5.2/32 is directly connected, Multilink1
D     192.168.9.0/24 [90/1343488] via 192.168.5.2, 16:02:07, Multilink1

R1# show ip eigrp interfaces
EIGRP-IPv4 Interfaces for AS(1)
                   Xmit Queue   PeerQ        Mean   Pacing Time   Multicast    Pending
Interface   Peers  Un/Reliable  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Mu1         1      0/0          0/0          1      0/8           50           0
Gi0/0       1      0/0          0/0          1      0/0           50           0
```

```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status                Protocol
Embedded-Service-Engine0/0 unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0         192.168.1.1     YES manual up                    up
GigabitEthernet0/1         unassigned      YES manual up                    up
Serial0/0/0                unassigned      YES manual up                    up
Serial0/0/1                unassigned      YES manual administratively down down
Serial0/1/0                unassigned      YES NVRAM  administratively down down
Serial0/1/1                unassigned      YES NVRAM  up                    up
Multilink1                 192.168.5.1     YES manual up                    up
```

Gehen wir das Listing von oben nach unten durch, sehen wir, dass das Interface *multilink 1* für eine ganze Reihe von Routen als ausgehendes Interface in der IPv4-Routing-Tabelle angegeben ist. Die beiden seriellen Interfaces hingegen erscheinen überhaupt nicht, denn sie haben keine IP-Adressen, und die Routing-Logik des Routers nutzt stattdessen das Multilink-Interface. Ähnlich listet der Befehl **show ip eigrp interfaces** Interfaces auf, für die EIGRP aktiviert ist. Hier erscheint *Mu1* (»Multilink 1«) in der Auflistung, nicht aber ein anderes der beiden seriellen Interfaces im MLPPP-Bündel. Beachten Sie abschließend, dass der Befehl **show ip interface brief** die beiden seriellen Interfaces *und* das Multilink-Interface angibt; aus der Ausgabe geht aber hervor, dass für die seriellen Interfaces keine IP-Adressen konfiguriert wurden (Eintrag **unassigned** in der Spalte **IP-Address**).

Jedes Multilink-Interface hat wie jedes andere Interface auch einen Leitungs- und einen Protokollstatus. Ist dieser Status *up/up*, dann geht das IOS davon aus, dass das Multilink-Interface funktioniert. Normalerweise impliziert dieser Status, dass mindestens eine physische Leitung in der MLPPP-Gruppe einsatzbereit ist – was nichts anderes bedeutet, als dass der Multilink auch bei Ausfall einer oder mehrerer (wenn auch nicht aller) physischen Leitungen verfügbar bleibt. Sie können die seriellen Interfaces in der Multilink-Gruppe jederzeit direkt mit den Befehlen überprüfen, die wir weiter oben im Kapitel bereits behandelt haben (**show controllers**, **show interfaces**). Außerdem gewähren die beiden Befehle in Listing P.9 einen Einblick in die Besonderheiten des MLPPP-Betriebs.

Listing P.9 Betriebsdetails einer MLPPP-Gruppe überprüfen

```
R1# show interfaces multilink 1
Multilink1 is up, line protocol is up
  Hardware is multilink group interface
  Internet address is 192.168.5.1/24
  MTU 1500 bytes, BW 3088 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open, multilink Open
  Open: IPCP, CDPCP, loopback not set
  Keepalive set (10 sec)
! Zeilen aus Platzgründen gekürzt

R1# show ppp multilink
```

```
Multilink1
  Bundle name: R2
  Remote Username: R2
  Remote Endpoint Discriminator: [1] R2
  Local Username: R1
  Local Endpoint Discriminator: [1] R1
  Bundle up for 16:50:33, total bandwidth 3088, load 1/255
  Receive buffer limit 24000 bytes, frag timeout 1000 ms
    0/0 fragments/bytes in reassembly list
    0 lost fragments, 96 reordered
    0/0 discarded fragments/bytes, 0 lost received
    0x654D7 received sequence, 0x654D5 sent sequence
  Member links: 2 active, 0 inactive (max 255, min not set)
    Se0/1/1, since 16:50:33
    Se0/0/0, since 16:23:16
No inactive multilink interfaces
```

Beachten Sie, dass der Befehl **show interfaces multilink 1** viele vertraute Details auflistet, darunter auch einige zum Multilink. Wir finden hier zunächst wie üblich den Leitungs- und den Protokollstatus (*up/up*). Zudem steht in der sechsten Zeile der Multilink-Status *Open*.

Abschließend sind in der Ausgabe des Befehls **show ppp multilink** die Links aufgeführt, die im jeweiligen Multilink-Bündel konfiguriert und ggf. aktiv sind. Im vorliegenden Fall sind die Interfaces S0/0/0 und S0/1/1 auf R1 aktiv (siehe die Hervorhebung am Ende des Listings). Der Timer zeigt, dass dies bereits seit etwas mehr als 16 Stunden der Fall ist. Das Auftauchen dieser beiden Interfaces in der Liste beweist nicht nur, dass die physischen Interfaces funktionieren, sondern auch, dass die MLPPP-Konfiguration beide Leitungen zur Multilink-Gruppe 1 hinzugefügt hat.

Problembehebung bei seriellen Verbindungen

Dieser letzte Hauptabschnitt erläutert, wie man die Ursachen von Problemen für die in diesem Kapitel angesprochenen Themen findet. Dieser Abschnitt soll keine Wiederholung der Ausführungen zur IP-Problembehebung in Teil II darstellen. Vielmehr soll er einige mögliche Symptome bei seriellen Verbindungen aufzeigen, wenn eine Subnetzfehlanpassung an den beiden Enden einer seriellen Verbindung auftritt, die verhindert, dass die Router Pakete über diese Verbindung übertragen können.

Mithilfe eines einfachen **ping**-Befehls lässt sich ermitteln, ob eine serielle Verbindung IP-Pakete weiterleiten kann oder nicht. Ein **ping** auf die andere serielle IP-Adresse des Routers – z. B. ein ausgeführtes **ping 192.168.2.2** für R1 in Abbildung P.11 (die für die HDLC- und PPP-Konfigurationsbeispiele verwendete Abbildung) – lässt erkennen, ob die Verbindung funktioniert oder nicht.

Wenn **ping** nicht funktioniert, könnte das Problem mit Funktionen in den Layern 1, 2 oder 3 zusammenhängen. Am besten grenzt man die betroffenen Layer ein, indem man die Interfacestatuscodes aus Tabelle P.4 untersucht.

Tabelle P.4 Interfacestatuscodes und typische Bedeutungen bei Fehlschlagen eines **ping**-Befehls

Line-Status	Protokollstatus	Wahrscheinliche Ursache/Layer
Administratively Down	Down	Interface shutdown
Down	Down	Layer 1
Up	Down	Layer 2
Up	Up	Layer 3

Der Prozess für die Verifizierung der seriellen Verbindung und das Troubleshooting sollte mit diesen drei Schritten beginnen:

Schritt 1: Senden Sie von einem Router aus einen **ping**-Befehl an die serielle IP-Adresse des anderen Routers.

Schritt 2: Schlägt der **ping**-Befehl fehl, so untersuchen Sie den Interfacestatus auf beiden Routern und prüfen Sie auf Probleme im Zusammenhang mit den in Tabelle P.4 aufgeführten Problembereichen.

Schritt 3: Auch wenn der **ping**-Befehl erfolgreich ist, sollten Sie überprüfen, ob Routing-Protokolle tatsächlich Routen über die Verbindung austauschen.

HINWEIS Die Interfacestatuscodes können Sie mit den Befehlen **show interfaces**, **show ip interface brief** und **show interfaces description** ermitteln.

Anschließend werden wir die spezifischen Aspekte untersuchen, die basierend auf den in Tabelle P.4 aufgeführten Kombinationen der Interfacestatuscodes zu überprüfen sind, wenn der **ping**-Befehl fehlschlägt.

Troubleshooting von Layer-1-Problemen

Die Interfacestatuscodes – oder kurz Interfacestatus – spielen eine wichtige Rolle bei der Eingrenzung der Hauptursache von Problemen bei seriellen Verbindungen. Tatsächlich können sich die Zustände an beiden Enden der Verbindung unterscheiden, weswegen es wichtig ist, zur Ermittlung des Problems beide Zustände zu kontrollieren.

Bei seriellen Verbindungen kann dies vorkommen, wenn z. B. ein Router sein serielles Interface mit dem Interfacesubbefehl **shutdown** administrativ deaktiviert hat. Wenn ein Router sein serielles Interface abschaltet, befindet sich der andere Router im Status *down/down* (Line-Status *down*, Protokollstatus *down*), vorausgesetzt, das Interface des anderen Routers ist nicht ebenfalls abgeschaltet. Die Lösung besteht einfach darin, für das Interface den Konfigurationsbefehl **no shutdown** zu konfigurieren.

Wenn ein serielles Interface den Line-Status *down* an beiden Enden der seriellen Verbindung aufweist (d. h., beide haben den Status *down/down*), dann weist das normalerweise auf ein Layer-1-Problem hin. Abbildung P.22 fasst die wichtigsten Ursachen dieses Status zusammen. In der Abbildung hat das serielle Interface von R2 keine Probleme. In der Mitte und links sehen Sie häufige Ursachen, die dazu führen können, dass sich das serielle R2-Interface im Status *down/down* befindet.

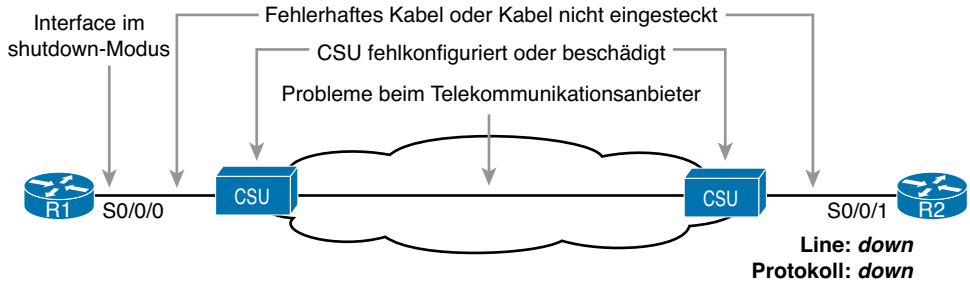


Abbildung P.22 Probleme, die dazu führen, dass Router R2 den Status **down/down** bekommt

Troubleshooting von Layer-2-Problemen

Probleme im Data Link Layer bei seriellen Verbindungen führen normalerweise dazu, dass bei mindestens einem Router der Status des seriellen Interface *up/down* ist. Der Line-Status (der erste Statuscode) ist mithin *up*, der zweite (also der Protokollstatus) ist *down*. Tabelle P.5 fasst diese Arten von Problemen zusammen.

Tabelle P.5 Wahrscheinliche Gründe für Data-Link-Layer-Probleme bei seriellen Verbindungen

Line-Status	Protokollstatus	Wahrscheinliche Ursache
Up	Ausfall an beiden Enden[1]	Fehlangepasste **encapsulation**-Befehle
Up	*down* am einen Ende, *up* am anderen	Deaktivierter Keepalive an dem Ende im Status *up* beim Einsatz von HDLC
Up	*down* an beiden Enden	Fehlgeschlagene PAP- bzw. CHAP-Authentifizierung

[1] In diesem Fall kann der Status von *up/up* nach *up/down* und wieder zu *up/up* usw. wechseln, weil der Router versucht, die Kapselung umzusetzen.

Das erste dieser beiden Probleme – eine Fehlanpassung zwischen den konfigurierten Data-Link-Protokollen – ist einfach zu erkennen und zu beheben. Der Befehl **show interfaces** gibt den Kapselungstyp in der siebten Ausgabezeile an, weswegen sich das Problem mit dessen Hilfe schnell erkennen lässt, wenn man ihn bei den beiden Routern anwendet. Alternativ überprüft man kurz in der Konfiguration – verbunden mit dem Wissen, dass HDLC die serielle Standardkapselung ist –, ob die Kapselungen zueinander passen. Die Lösung ist einfach: Konfigurieren Sie einen der beiden Router so um, dass er zum Befehl **encapsulation** auf dem anderen Router passt.

Die anderen beiden Hauptursachen erfordern eine etwas ausführlichere Beschreibung, um das Problem zu begreifen und feststellen zu können, ob dies wirklich die Hauptursachen sind. Die nächsten beiden Abschnitte enthalten diese Beschreibungen.

Keepalive-Ausfall

Durch *Keepalives* können Router erkennen, wann eine Verbindung nicht mehr funktioniert. Wenn der Router davon ausgeht, dass die Verbindung nicht mehr funktioniert, kann er das Interface abschalten und das Routing-Protokoll nimmt dann, falls vorhanden, eine alternative IP-Route.

Die Keepalive-Funktion auf einem Interface bewirkt bei Routern (standardmäßig) das Senden von Keepalive-Nachrichten in einem konfigurierten Keepalive-Intervall, das per Default zehn Sekunden beträgt. Auf einem seriellen Link zwischen R1 und R2 sendet R1 alle zehn Sekunden eine Keepalive-Nachricht und R2 erwartet auch, alle zehn Sekunden eine solche Nachricht zu empfangen. Wenn R2 für eine bestimmte Anzahl aufeinanderfolgender Keepalive-Intervalle (meistens drei oder fünf) keine Keepalive-Meldungen erhält, geht der Router davon aus, dass R1 ausgefallen ist, und ändert den Status der Verbindung entsprechend in *up/down*. Der Keepalive-Vorgang verläuft in beide Richtungen, d. h., nicht nur sendet R1 Keepalives an R2, die von diesem auch erwartet werden, sondern R2 sendet seinerseits Keepalives an R1, die wiederum von diesem Router erwartet werden.

Eine Keepalive-Fehlanpassung tritt auf, wenn Keepalives auf dem einen Router aktiviert sind, auf dem anderen jedoch nicht. Eine solche Kombination ist fehlerhaft und darf nicht verwendet werden. Beachten Sie, dass eine Fehlanpassung bei den Keepalives nur für den Abbruch von HDLC-Verbindungen sorgt; das Keepalive-Feature von PPP verhindert dieses Problem. In Abbildung P.23 sehen Sie ein solches Beispiel, wo bei R1 unter Verwendung von HDLC die Keepalives fälschlicherweise deaktiviert sind.

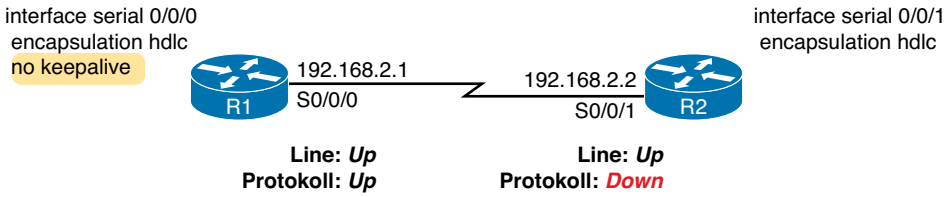


Abbildung P.23 Probleme, die dazu führen, dass Router R2 den Status **up/down** annimmt

Beachten Sie, dass das Router-Interface, auf dem Keepalives deaktiviert sind, im Status *up/up* verbleibt. Im Szenario aus Abbildung P.23 funktioniert das Interface von R2 nicht, weil

- R1 keine Keepalive-Nachrichten sendet, da Keepalives deaktiviert wurden.
- R2 weiterhin erwartet, Keepalive-Nachrichten zu erhalten, weil Keepalives aktiviert sind.

Sie können die Keepalive-Einstellung wahlweise durch einen Blick auf die Konfiguration oder mithilfe des Befehls **show interfaces** überprüfen. In den Listings in diesem Kapitel sind mehrere Beispiele für den Befehl **show interfaces** vorhanden, die alle den Text *Keepalive set (10 second)* enthalten; das bedeutet, dass Keepalives im 10-Sekunden-Abstand aktiviert sind. R1 würde in diesem Fall den Text *Keepalive not set* ausgeben.

Authentifizierungsfehler bei PAP und CHAP

Wie bereits erwähnt, führt ein fehlgeschlagener PAP- oder CHAP-Authentifizierungsprozess dazu, dass beide Router-Interfaces in den Status *up/down* zurückfallen. Wie die Listings P.6 und P.7 gezeigt haben, können Sie den Status des PPP-Authentifizierungsvorgangs mit den Befehlen **show interfaces** und **show ppp all** genauer untersuchen. Sie können dabei die Ursache für den Status *up/down* des Interface eingrenzen und die PPP-Authentifizierung als Hauptursache einbeziehen oder ausschließen.

Eine weitere Methode für ein ausführliches Troubleshooting der PPP-Authentifizierung ist die Verwendung des Befehls **debug ppp authentication**.

CHAP verwendet wie weiter vorne in Abbildung P.14 gezeigt den Austausch dreier Nachrichten. Dabei fließt ein Nachrichtensatz zur Authentifizierung standardmäßig in beide Richtungen. Wenn Sie das Debugging aktivieren und die Leitung ab- und dann wieder aufbauen, werden Debugmeldungen angezeigt, die dem 3-Schritte-Austausch entsprechen. Bei fehlgeschlagener Authentifizierung erscheint eine Fehlermeldung an der Stelle, an der der Vorgang abbricht. Hieraus können Sie möglicherweise schließen, was genau korrigiert werden muss.

Listing P.10 zeigt die drei zusammengehörigen Debugmeldungen beim Wiederaufbau des Links. Das Netzwerk stellt eine Verbindung des Interface S0/0/0 auf R1 mit einem Router R2 her. Im Listing sehen wir die drei relevanten Debugmeldungen, die wir aus der Gesamtzahl von mehreren Dutzend Fehlermeldungen herausgezogen haben. Rechnen Sie also damit, nach solchen Meldungen suchen zu müssen. Allerdings sind in der Ausgabe die folgenden wichtigen Teile des Vorgangs (vgl. Abbildung P.14 weiter vorne) hervorgehoben:

1. Das **O** bezeichnet eine Ausgabe (»Output«), d. h., dieser lokale Router R1 hat eine Challenge-Nachricht versendet. Die Angabe **from R1** am Ende der Debugmeldung gibt an, woher die Nachricht stammt.
2. Das **I** bezeichnet eine Eingabe (»Input«), d. h., dieser lokale Router R1 hat eine Response-Nachricht empfangen. Beachten Sie hier die Angabe **from R2** am Zeilenende.
3. **O FAILURE** gibt an, dass R1 eine Ausfallmeldung versendet und damit R2 davon in Kenntnis setzt, dass der Authentifizierungsvorgang fehlgeschlagen ist.

Listing P.10 Debug-Nachrichten auf R1, die den Fehlschlag bei CHAP bestätigen

```
R1# debug ppp authentication
PPP authentication debugging is on
! Zeilen aus Platzgründen gekürzt
*Nov 18 23:45:48.820: Se0/0/0 CHAP: O CHALLENGE id 1 len 23 from "R1"
*Nov 18 23:45:48.820: Se0/0/0 CHAP: I RESPONSE id 1 len 23 from "R2"
*Nov 18 23:45:48.820: Se0/0/0 CHAP: O FAILURE id 1 len 25 msg is “Authentication
  failed”
```

Einem **debug**-Befehl können wir möglicherweise Informationen zum Problem entnehmen, aber leider geht aus der Ausgabe nicht immer hervor, welcher Befehl ganz konkret fehlkonfiguriert ist. Im vorliegenden Fall lässt die Tatsache, dass beide Router mindestens eine CHAP-Nachricht versendet haben, den Schluss zu, dass beide Router-Interfaces Frames versenden können und dass auf beiden CHAP aktiviert ist. Es sieht eher so aus, als habe R1 den von R2 übermittelten

Passwort-Hash zurückgewiesen. Beachten Sie, dass dieses Beispiel dadurch erstellt wurde, dass ich für den Befehl **username** ein fehlerhaftes Passwort festgelegt habe: Der CHAP-Prozess hat also funktioniert, die Authentifizierung hingegen logischerweise nicht

Troubleshooting von Layer-3-Problemen

In diesem Kapitel wurde bereits gesagt, dass zur Behebung von Problemen mit seriellen Verbindungen der beste Startpunkt darin besteht, einen **ping**-Befehl an die IP-Adressen des Routers am anderen Ende der seriellen Verbindung zu senden. Interessanterweise kann sich die serielle Verbindung im Status *up/up* befinden, doch der **ping**-Befehl kann aufgrund einer Fehlkonfiguration von Layer 3 trotzdem fehlschlagen. In manchen Fällen funktioniert der **ping**-Befehl, doch können die Routing-Protokolle unter Umständen die Routen nicht austauschen. Die folgende Kurzbeschreibung untersucht die Symptome, die sich, je nachdem, ob HDLC oder PPP verwendet werden, und abhängig von der Hauptursache unterscheiden können.

Betrachten wir zunächst eine HDLC-Verbindung, bei der der Physical und der Data Link Layer einwandfrei funktionieren. In diesem Fall haben die Interfaces beider Router den Status *up/up*. Befinden sich jedoch die auf den seriellen Interfaces der beiden Router konfigurierten IP-Adressen in unterschiedlichen Subnetzen, dann schlägt der **ping**-Befehl an die IP-Adresse am anderen Ende der Verbindung fehl, weil es für die Router keine passende Route gibt. Lautete die IP-Adresse des seriellen Interfaces von R1 in Abbildung P.17 beispielsweise 192.168.2.1 und würde die IP-Adresse von R2 unter Beibehaltung der Subnetzmaske /24 auf 192.168.3.2 (statt 192.168.2.2) umgestellt, dann würden die beiden Router über angeschlossene Routen in verschiedene Subnetze verfügen. Sie hätten dann keine passende Route zur seriellen IP-Adresse des anderen Routers.

Das Auffinden und Beheben von Subnetzfehlanpassungen bei HDLC-Verbindungen ist relativ einfach. Sie ermitteln das Problem, indem Sie den normalen ersten Schritt ausführen: einen **ping**-Befehl an die IP-Adresse auf der anderen Seite der Verbindung senden, was misslingt. Wenn die Statuscodes der Interfaces beider Router *up/up* sind, ist das Problem wahrscheinlich dieses fehlangepasste IP-Subnetz.

Bei PPP-Verbindungen mit der gleichen Fehlkonfiguration von IP-Adresse und Maske ist das Senden des **ping**-Befehls an die IP-Adresse des anderen Routers erfolgreich. Jedoch verhindert die Fehlkonfiguration des IP-Subnetzes weiterhin, dass sich EIGRP- und OSPF-Nachbarschaftsbeziehungen bilden. Also bleibt es eine gute Idee, die Regeln zu befolgen und beide IP-Adressen der seriellen Interfaces ins gleiche Subnetz zu legen.

PPP sorgt dafür, dass der **ping**-Befehl innerhalb des falsch konfigurierten Subnetzes funktioniert, indem für die IP-Adresse des anderen Routers eine Hostroute mit der Präfixlänge /32 eingefügt wird. Das gehört zur Arbeit des IP Control Protocol. Listing P.11 zeigt genau dieses Szenario.

HINWEIS Eine Route mit dem Präfix /32, die einen einzelnen Host repräsentiert, heißt *Hostroute*.

Listing P.11 Erfolgreicher **ping**-Befehl über eine serielle Verbindung trotz fehlangepasster Subnetze

```
R1# show ip route
! Legende aus Platzgründen weggelassen
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/0
L        192.168.1.1/32 is directly connected, GigabitEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, Serial0/0/0
L        192.168.2.1/32 is directly connected, Serial0/0/0
      192.168.3.0/32 is subnetted, 1 subnets
C        192.168.3.2 is directly connected, Serial0/0/0

R1# ping 192.168.3.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Die erste hervorgehobene Zeile im Listing zeigt die normale angeschlossene Route der seriellen Verbindung in das Netzwerk 192.168.2.0/24. R1 nimmt aufgrund der konfigurierten IP-Adresse (192.168.2.1/24) an, dass dieses Subnetz das an S0/0/0 angeschlossene Subnetz ist. Die zweite hervorgehobene Zeile zeigt die Hostroute, die von PPP für die neue serielle IP-Adresse von R2 (192.168.3.2) erstellt wurde. (R2 verfügt über eine ähnliche Route nach 192.168.2.1/32, der seriellen IP-Adresse von R1.) Insofern verfügen beide Router über eine Route, die es ihnen gestattet, Pakete an die IP-Adresse auf der anderen Seite der Verbindung weiterzuleiten, auch wenn sich die Adresse des anderen Routers in einem anderen Subnetz befindet. Diese zusätzliche Hostroute macht es möglich, dass der an die andere Seite der seriellen Verbindung gerichtete **ping**-Befehl trotz der Tatsache funktioniert, dass sich die Adressen auf beiden Seiten in unterschiedlichen Subnetzen befinden.

Tabelle P.6 fasst das Verhalten bei HDLC- und PPP-Verbindungen zusammen, wenn die IP-Adressen an den beiden Enden nicht im selben Subnetz liegen, aber keine anderen Probleme vorhanden sind.

Tabelle P.6 Zusammenfassung der Symptome bei fehlangepassten Subnetzen an seriellen Verbindungen

Symptome, wenn sich IP-Adressen an einer seriellen Verbindung in verschiedenen Subnetzen befinden	HDLC	PPP
Funktioniert ein **ping**-Befehl an die IP-Adresse des seriellen Interface des anderen Routers?	Nein	Yes
Können Routing-Protokolle Routen über die Verbindung austauschen?	Nein	Nein

Befehlsreferenz

Die Tabellen P.7 und P.8 listen die in diesem Kapitel verwendeten Konfigurations- bzw. Überprüfungsbefehle auf. Eine einfache Wiederholungsübung besteht darin, die linke Spalte einer Tabelle abzudecken, die rechte Spalte zu lesen und sich ohne nachzusehen den entsprechenden Befehl in Erinnerung zu rufen. Wiederholen Sie die Übung dann, indem Sie nun die rechte Spalte abdecken und dann überlegen, was der jeweilige Befehl bewirkt.

Tabelle P.7 Referenz der Konfigurationsbefehle aus Anhang P

Befehl	Beschreibung
encapsulation {**hdlc** | **ppp**}	Interfacesubbefehl, der das serielle Data-Link-Protokoll festlegt
[**no**] **shutdown**	Deaktiviert (**shutdown**) oder aktiviert (**no shutdown**) administrativ das Interface, in dessen Modus der Befehl eingegeben wird.
clock rate *rate*	Serieller Interfacesubbefehl, der für ein Interface mit DCE-Kabel die Taktrate in Bit/s festlegt
bandwidth *datenrate-in-kbit/s*	Interfacesubbefehl, der dem Router eine Verbindungsgeschwindigkeit in Kbit/s vorgibt. Hat aber keine Auswirkungen auf die eigentliche Geschwindigkeit.
description *text*	Interfacesubbefehl, mit dem eine Interfacebeschreibung konfiguriert werden kann
ppp authentication {**pap** | **chap**}	Interfacesubbefehl, der nur PAP oder nur CHAP aktiviert
username *name* **password** *secret*	Globaler Befehl, der das Passwort festlegt, dessen Verwendung der Router bei der Authentifizierung mit dem aufgeführten Hostnamen erwartet
ppp pap sent-username *name* **password** *secret*	Interfacesubbefehl, mit dem die Benutzername-Passwort-Kombination definiert wird, die über diesen Link bei Verwendung der PAP-Authentifizierung versendet wird
interface multilink *nummer*	Hiermit wird ein Multilink-Interface erstellt und der Benutzer wechselt in den Interfacekonfigurationsmodus für dieses Interface.
ppp multilink	Interfacesubbefehl zur Aktivierung der MLPPP-Funktionen
ppp multilink group *nummer*	Interfacesubbefehl, der das Interface mit einem bestimmten Multilink-Interface und einer Multilink-Gruppe verknüpft

Tabelle P.8 Referenz der EXEC-Befehle aus Anhang P

Befehl	Beschreibung
show interfaces [*typ nummer*]	Führt Statistiken und Details zur Interface-Konfiguration auf, darunter den Kapselungstyp.
show interfaces [*typ nummer*] **description**	Zeigt je Interface eine Informationszeile an (falls das Interface darin eingebunden ist, nur eine Ausgabezeile insgesamt) mit dem Interfacestatus und einer für das jeweilige Interface konfigurierten Beschreibung.
show ip interface brief	Gibt eine Zeile pro Interface mit IP-Adresse und Interfacestatus aus.
show controllers serial *nummer*	Listet auf, ob ein Kabel mit dem Interface verbunden ist, und wenn ja, ob es sich um ein DTE- oder DCE-Kabel handelt.
show ppp multilink	Listet ausführliche Statusinformationen zu allen auf dem Router konfigurierten PPP-Multilink-Gruppen auf.
show ppp all	Listet genau eine Zeile mit Statusinformationen einschließlich Steuerprotokollstatus und IP-Adresse des Peer-Routers für jeden PPP-Link auf dem Router auf.
debug ppp authentication	Generiert Nachrichten für jeden Schritt im PAP- oder CHAP-Authentifizierungsvorgang.
debug ppp negotiation	Generiert **debug**-Meldungen für die LCP- und NCP-Verhandlungsnachrichten, die zwischen den Geräten ausgetauscht werden.