

## Umstellung einer statischen Routinginfrastruktur auf eine dynamische



Name	Datum	Version
Sebastian Wieseler	03.07.2010	1.0

<sup>1</sup> Bildquelle: Nationaal Archief / Spaarnestad Photo, SFA003008498. Signing post at Schiphol airport.  
<http://www.flickr.com/photos/nationaalarchief/4026230486/>. 1937

## Abstract

Dieser Anwenderbericht soll einen Weg aufzeigen von einer statischen Routinginfrastruktur auf eine dynamische umzustellen. Dabei wird im Vorfeld das Szenario erläutert, sowie die einzusetzenden Verfahren erklärt.

## Was ist Routing?

Felix von Leitner definiert in [Lei00] Routing als „eine Information, die einem Gerät sagt, wie es ein Paket in ein bestimmtes Ziel-Subnetz senden kann.“ Weiterhin definiert er eine Route als asymmetrisch, „wenn die Rückroute einen anderen Pfad benutzt“.

Router empfangen dabei Pakete auf einem ihrer Interfaces und leiten sie, basierend auf den Daten im IP Header, dann auf einem (oder mehrere, falls es sich hierbei um Multicast-Pakete handelt) anderen Interface weiter. Dabei erfolgt das Weiterleiten Hop-by-Hop. Der IP Header bleibt dabei relativ unverändert auf dem Weg zwischen den jeweiligen Routers. Routing beschreibt nun die Findung des next-Hop Routers, der dann die Weiterleitung der Pakete in Richtung des Zielnetzwerkes übernimmt. [Moy98, S.9].

## Das Szenario

Auf einem Campus besitzen wir zwei unterschiedliche Subnetze, die zwischen zwei Core-Switches durch Routing<sup>2</sup> verbunden werden.

Es existiert bisher eine statische Route zwischen diesen beiden Switches. Im Laufe der Zeit wurde jedoch eine weitere Verbindung zwischen diesen beiden Netzen geschaffen. Redundante Wege können durch statische Routing-Protokolle jedoch schlecht abgebildet werden.

Die Zielstellung sieht nun vor durch Einführung intelligenter Metriken den besseren beider Wege zu finden bzw. die Last gleichmäßig zu verteilen (load sharing). Wir werden deshalb ein dynamisches Routing-Protokoll einführen: OSPF.

## Warum OSPF?

Das *Open Shortest Path First* Verfahren ist im RFC 2328<sup>3</sup> festgelegt (obsolet: RFC 1247<sup>4</sup> von 1991). Es basiert auf dem *Shortest Path First Algorithmus* von Edsger W. Dijkstra.

Das heißt „[j]eder Router akkumuliert einen Graphen aus Routen [...] Die Nachbarn werden mit dem OSPF Hello protocol gefunden, eine Art ping, das auch als keepalive benutzt wird.“ [Lei00]

---

<sup>2</sup> Routing findet auf Layer 3 im OSI-Schichtenmodell statt

<sup>3</sup> RFC 2328: <http://tools.ietf.org/html/rfc2328>

<sup>4</sup> RFC 1247: <http://tools.ietf.org/html/rfc1247>

Gegenüber *RIP*<sup>5</sup> hat OSPF diverse Vorteile; so existiert nunmehr keine Beschränkung des Hop-Counts (vergl. *RIP* unterstützt max. 15 Hops). Weiterhin propagiert OSPF Änderungen sofort und nicht periodisch. Hierfür benutzt es IP Multicasting. Desweiteren erlaubt es besseres Load Balancing, sowie Routing Authentifikation mit diversen Passwort-Authentifizierungsmechanismen. [Cisco05]

OSPF setzt auf einen Link-State Algorithmus; *RIP* hingegen basiert auf einem Distanz Vektor Algorithmus. Letzterer hat den Nachteil, dass eine inkrementelle, verteilte Berechnung statt findet. Unter Nutzung von einem Link-State Algorithmus teilt jeder Router seine lokale Umgebung seinen Nachbarn mithilfe von LSAs<sup>6</sup> mit. Diese beinhalten alle aktiven Links zu seinem lokalen IP Netzwerk und seinen direkten Nachbarn mit entsprechenden Kosten<sup>7</sup> der Links. Es resultiert daraus eine Datenbank aus sämtlichen LSAs, die in allen Routern gleich ist, welche das momentane Netzwerk komplett beschreibt. Basierend auf diesem Gesamtgraphen kann nun jeder Router eine kürzeste Wegeberechnung (normalerweise Dijkstra) durchführen. [Moy98, S.39]

### Die Grundlagen: Der Algorithmus von Dijkstra

Der Shortest Path First Algorithmus ist aus der Klasse der Greedy-Algorithmen. Er „dient der Berechnung eines kürzesten Pfades zwischen einem Startknoten und einem oder mehreren beliebigen Knoten in einem kantengewichteten Graphen.“ [WP02+:Dijkstra] Dabei liefert er immer die optimale Lösung.

Informell schreibt [WP02+:Dijkstra] den Algorithmus wie folgt nieder:

- i) Weise allen Knoten die beiden Eigenschaften „Distanz“ und „Vorgänger“ zu. Initialisiere die Distanz im Startknoten mit 0 und in allen anderen Knoten mit  $\infty$ .
- ii) Solange es noch unbesuchte Knoten gibt, wähle darunter denjenigen mit minimaler Distanz aus und
  - i) Speichere, dass dieser Knoten schon besucht wurde
  - ii) Berechne für alle noch unbesuchten Nachbarknoten die Summe des jeweiligen Kantengewichtes und der Distanz im aktuellen Knoten
  - iii) Ist dieser Wert für einen Knoten kleiner als die dort gespeicherte Distanz, aktualisiere sie und setze den aktuellen Knoten als Vorgänger

### OSPF: Get inside

Sobald ein Router mit einem seiner Nachbarn ein Hello-Paket<sup>8</sup> ausgetauscht hat teilt er diese Information seiner Umgebung mit. Dies tut er mit den schon erwähnten LSAs. Diese Pakete beginne mit einem Link-Status Header. LSAs haben einen automatischen Verfall nach einer Stunde. Das bedeutet, dass jeder Router, dessen Nachbarrouten noch valide sind, nach dreißig Minuten sein LSA erneuern wird. [Tho02, S.133]

---

<sup>5</sup> Routing Information Protocol

<sup>6</sup> Link-State Advertisement

<sup>7</sup> Kosten stellen die entsprechenden Metriken der Links bzw. die Anzahl der Hops auf einem Pfad dar

<sup>8</sup> Hello-Pakete werden auf einem Interface alle 10 Sekunden ausgesendet (konfigurierbar via **HelloInterval**)

OSPF Pakete werden direkt über den IP Netzwerk Layer transportiert, ohne zu Hilfenahme von TCP oder UDP. Im IP-Header wird dazu die *TTL*<sup>9</sup> auf eins gestellt. Dies verhindert, dass falsch konfigurierte Router Pakete fälschlicherweise an andere weiterleiten. Das Feld der Ziel IP-Adresse wird auf die IP des Nachbarrouters gesetzt bzw. auf eine OSPF Multicast-Adresse (AllSPFRouters 224.0.0.5, AllDRouters 224.0.0.6). [Moy98, S.85]

Netzwerksegmente, die durch Router miteinander verbunden sind, werden in OSPF mit *Areas* abgebildet. Sie werden hierbei durch eine 32-bit ID dargestellt. Jede Area hat hierbei eine eigene Link-State-Datenbank, welche Router-LSAs und Netzwerk-LSAs enthält. Sie gibt Auskunft über die Verknüpfung der Router untereinander und Abhängigkeiten zu anderen Netzwerksegmenten. Das Routing innerhalb eines solchen Segmentes ist flach, mit der Information, dass jeder Router genau weiß welche Netzwerksegmente sich in dieser Area befinden. Eine genauere Kenntnis der internen Topologie innerhalb einer Area wird jedoch nach ausserhalb verborgen. Das bedeutet, dass Router- und Netzwerk-LSAs nicht über Area-Grenzen hinaus weitergeleitet werden. Router, die Konnektivität zu zwei oder mehreren Areas besitzen werden *ABRs*<sup>10</sup> genannt.

Wenn man eine OSPF Domäne in Areas untergliedert, müssen diese jedoch direkt zu einer speziellen Area angegliedert sein: der OSPF *Backbone Area* (ID 0.0.0.0). Die Verknüpfung zu dieser muss aber nicht jedem Fall ein physischer Link sein. Es bietet sich hierbei eine logische Verknüpfung an, die *Virtual Links* in OSPF. Diese bieten die Möglichkeiten summary-LSAs durch andere Areas durch zu tunneln. Hinzu kommen zwei Typen von Area, die man weiterhin in OSPF definieren kann: Unter einer *Stub Area*, versteht man einen Bereich in den keine AS-external-LSAs weitergeleitet werden. Der Weg hinaus besteht hauptsächlich aus statischen Default-Routen, die am ABR terminieren. Eine Unterstützung von Virtual Links ist ebenfalls nicht vorhanden, was lediglich eine Position an den Außenseiten einer OSPF-Infrastruktur ermöglicht. Eine Erweiterung der Stub Areas bieten die *NSSAs*<sup>11</sup>. Hierbei ist es doch wieder möglich externe Routing-Informationen (von anderen Protokollen) ins OSPF-Netz zu importieren. Die ABRs stellen hierbei einen one-way-Filter der externen Informationen zu anderen Areas dar (aber nicht umgekehrt). [Moy98, S.122ff]

## OSPF: Allgemeine Konfigurationsmöglichkeiten

[Moy98, S.155f] stellt wesentliche Parameter der OSPF *MIBs*<sup>12</sup> heraus. Es findet eine Untergliederung in 12 Gruppen statt, welche insgesamt 99 Variablen enthalten. 61 davon sind lediglich read-only Variablen. Um einen Router in eine OSPF Domäne zu integrieren benötigt dieser eine IP-Adresse auf seinem OSPF-Interface. Diese wird konfiguriert über die Variable **ospfIfIpAddress**. Die Kosten dieses Interfaces wird über **ospfIfMetricValue** konfiguriert. Um einen Router einer bestimmten Area hinzuzufügen benötigt dieser eine **ospfIfAreaId**. Für einen Borderline-Router benötigt man zusätzliche Einträge in der **ospfAreaAggregateTable**, die zur Aggregation verschiedener Areas benutzt wird. Die virtuellen Links können über die **ospfVirtIfTable** gemanagt werden.

---

<sup>9</sup> Time to Live: Dieses Feld im IP-Header wird von jedem Router dekrementiert werden auf seinem Weg zum Zielknoten

<sup>10</sup> area border routers

<sup>11</sup> not-so-stubby areas

<sup>12</sup> Management Information Base: Informationen, welche über ein Netzwerk-Management-Protokoll abgefragt werden können (z.B. das Simple Network Management Protocol — SNMP)

## Die Ausgangslage

Zu Beginn finden wir folgende Router-Konfigurationen vor (Beispiel nach [Par98, S.206]):

Router R1 Konfiguration	Router R2 Konfiguration
hostname R1	hostname R2
interface Loopback 0	interface Loopback 0
ip address 172.16.1.1 255.255.255.0	ip address 172.16.4.1 255.255.255.0
interface Loopback 0	interface Loopback 0
ip address 172.16.2.1 255.255.255.0	ip address 172.16.5.1 255.255.255.0
interface Ethernet0	interface Ethernet0
ip address 172.16.3.1 255.255.255.09	ip address 172.16.3.2 255.255.255.09

## Die Konfiguration

Für die Konfiguration der OSPF Optionen werden wir den Cisco-Guide [Cisco05] verwenden. Als erstes müssen wir auf den Routern OSPF aktivieren. Das tun wir wie folgt<sup>13</sup>:

```
R1(config)#router ospf ?
<1-65535>          Process ID
R1(config)#router ospf 100
```

Danach werden wir die Areas<sup>14</sup> auf den Interfaces konfigurieren:

```
R1(config-router)#network ?
A.B.C.D Network number
R1(config-router)#network 172.16.1.1 ?
A.B.C.D OSPF wildcard bits
R1(config-router)#network 172.16.1.1 0.0.0.0 ?
Area Set the OSPF area ID
R1(config-router)#network 172.16.1.1 0.0.0.0 area 0
<0-4294967295>    OSPF area ID as a decimal value
A.B.C.D          OSPF area ID in IP address format
R1(config-router)#network 172.16.1.1 0.0.0.0 area 0
```

<sup>13</sup> <process-id>: Lokaler numerischer Wert (Möglichkeit mehrere OSPF-Instanzen auf einem Router zu verwenden)

<sup>14</sup> Link-Status Updates werden lediglich auf den Interfaces, die zu einer Area gehören, versendet. Dadurch *flutet* man nicht Informationen zu Routern, die sowieso keinen direkten Pfad zu diesem Netzabschnitt (Area) besitzen.

Nun soll Router 1 noch seine weiteren Netzwerke bekannt geben:

```
R1(config-router)#network 172.16.2.1 0.0.0.0 area0
R1(config-router)#network 172.16.3.1 0.0.0.0 area0
```

Für Router 2 verwenden wir eine analoge Konfiguration:

```
R2(config)# router ospf 100
R2(config-router)#network 172.16.3.2 0.0.0.0 area0
R2(config-router)#network 172.16.4.1 0.0.0.0 area0
R2(config-router)#network 172.16.5.1 0.0.0.0 area0
```

OSPF ist nun auf beiden Routern R1 und R2 aktiviert. Dies kann mit einem Blick in ihre Routingtabellen verifiziert werden:

```
R2# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile,
B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
OSPF inter area, N1 - OSPF NSSA external type 1, N2 -
OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, i - IS-IS, su - IS-IS
summary, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
IS-IS inter area, * - candidate default, U - per-user
static route, o - ODR, P - periodic downloaded static
route

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 5 subnets,
2 masks

C 172.16.4.0/24 is directly connected, Loopback0
C 172.16.5.0/24 is directly connected, Loopback1
O 172.16.1.1/32 [110/2] via 172.16.3.1, 00:02:49,
FastEthernet0/0
O 172.16.2.1/32 [110/2] via 172.16.3.1, 00:02:49,
FastEthernet0/0
C 172.16.3.0/24 is directly connected, FastEthernet0/0
```

```

R1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile,
B - BGP, D - EIGRP, EX - EIGRP external, O - OSPF, IA -
OSPF inter area, N1 - OSPF NSSA external type 1, N2 -
OSPF NSSA external type 2, E1 - OSPF external type 1,
E2 - OSPF external type 2, i - IS-IS, su - IS-IS
summary, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
IS-IS inter area, * - candidate default, U - per-user
static route, o - ODR, P - periodic downloaded static
route

Gateway of last resort is not set

 172.16.0.0/16 is variably subnetted, 5 subnets,
 2 masks
O 172.16.5.1/32 [110/2] via 172.16.3.2, 00:03:40,
  FastEthernet8/1
O 172.16.4.1/32 [110/2] via 172.16.3.2, 00:03:40,
  FastEthernet8/1
C 172.16.1.0/24 is directly connected, Loopback0
C 172.16.2.0/24 is directly connected, Loopback1
C 172.16.3.0/24 is directly connected, FastEthernet8/1

```

Router R1 und R2 haben dabei drei direkt angeschlossene Netzwerke/drei direkte Routen (C) und zwei Routen, welche sie über OSPF gelernt haben (O).

Die Kosten für das OSPF Netzwerk können berechnet werden, indem man die Bandbreiten der Links auf beiden Seiten addiert und danach durch 100.000.000 dividiert. [Par98, S.210] nimmt als Ethernet-Bandbreite von R2 100.000.000 an plus 80.000.000 für die Loopback-Bandbreite des Routers R1. Nun erhält man die Kosten der Route von 1,8 — gerundet auf 2.

Die Bandbreite des Ethernet Links kann dabei wie folgt ermittelt werden:

```

R2# show interface FastEthernet 8/1
FastEthernet8/1 is up, line protocol is up
Hardware is cyBus FastEthernet Interface, address is
0000.0ca5.d402 (bia 0000.0ca5.d402)
Internet address is 172.16.3.1/24
MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec, rely
255/255, load 1/255
Encapsulation ARPA, loopback not set, keepalive set (10
sec)
Half-duplex, 100Mb/s, 100BaseTX/FX
ARP type: ARPA, ARP Timeout 04:00:00
...

```

## OSPF Authentifikation

Standardmäßig besitzt OSPF eine Null-Authentifikation. Das bedeutet, dass Routing-Informationen nicht authentifiziert werden.

Eine andere Möglichkeit bietet eine symmetrische Verschlüsselung, bei der alle Router in einer Area einen eigenen Schlüssel verwenden. Diese Art der Kommunikation ist leicht anfällig für Abhörattacken. Sie wird wie folgt konfiguriert:

```
Router (config)# interface ...
Router (config-if)# ip address ...
Router (config-if)# ip ospf authentication-key mykey
Router (config-if)# exit
Router (config)# router ospf 100
Router (config-router)# network ... area 0
Router (config-router)# area 0 authentication
```

Eine weitere Möglichkeit bietet die Message Digest Authentifikation. Hierbei wird der Key nicht via Plaintext ausgetauscht und eine Sequenznummer wird eingeführt, um vor Replay-Attacken zu schützen. Bei einem Wechsel des Schlüssels auf einem Interface wird OSPF automatisch solange Pakete mit altem und neuen Schlüssel versenden, bis alle Nachbarrouter ihr Schlüsselmaterial entsprechend aktualisiert bekommen haben. In der Konfiguration sieht es wie folgt aus:

```
Router (config)# interface ...
Router (config-if)# ip address ...
Router (config-if)# ip ospf message-digest-key
    10 md5 mykey
Router (config-if)# exit
Router (config)# router ospf 100
Router (config-router)# network ... area 0
Router (config-router)# area 0 authentication
    message-digest
```



## OSPF Troubleshooting

Falls es mal nötig werden sollte, gibt [WSR05, S.184ff] Hinweise wo man mit dem Troubleshooting von OSPF beginnen sollte. Die Ausgabe der Nachbar-Routern, die über OSPF gesehen werden, erfolgt dabei mit:

```
Router# show ip ospf neighbor
```

Wenn diese Liste leer ist, ist eine mögliche *Adjazenz*<sup>15</sup> zu einem weiteren Router nicht vorhanden. Das Debugging dieses Fehlers wird gestartet mit:

```
Router# debug ip ospf adjacency
```

Das HelloInterval, DeadInterval, IP-Subnetz und der Link-Typ müssen auf beiden (Kanten)Seiten identisch konfiguriert sein, sonst ist eine Kommunikation zweier Router nicht vollständig gegeben.

Ein anderer erörterter Fehler ist, dass der Status der OSPF-Verbindung zwischen EXCHANGE und INIT hin- und her wechselt. Ersteres bedeutet, dass ein Datenbanktausch initialisiert wird. Es werden hierbei Hello-Pakete zwischen den Routern ausgetauscht, jedoch keine Datenbank-Pakete. Dies könnte daran liegen, dass 64 Bytes ausreichend sind für Hello-Pakete. Ein Mismatch der *MTU*<sup>16</sup>-Größe könnte jedoch dazu führen, dass größere Pakete nicht ordnungsgemäß zugestellt werden können. Daher wechselt der Verbindungsstatus hierbei kontinuierlich zwischen EXCHANGE und INIT.

## OSPF Implementierungen

Als weiterführende Literatur zur OSPF Implementierung (GNU/Linux ...) empfiehlt sich das Studium von [Moy01].

---

<sup>15</sup> Beziehung zwischen Knoten. Adjazente Knoten besitzen eine gemeinsame Kante in einem ungerichteten Graphen

<sup>16</sup> Maximum Transmission Unit: maximale unfragmentiert übertragene Datenmenge

## ZITIERTE WERKE

[Cisco05]: Cisco Systems. *OSPF Design Guide*, Document ID: 7039, 2005.  
[http://www.cisco.com/en/US/tech/tk365/technologies\\_white\\_paper09186a0080094e9e.shtml](http://www.cisco.com/en/US/tech/tk365/technologies_white_paper09186a0080094e9e.shtml)

[Lei00]: Felix von Leitner. *Routing in IP Netzen*, 2000.  
<http://www.fefe.de/routing/routing.pdf>

[Moy01]: John T. Moy. *OSPF: Complete Implementation*. Bosten: Addison-Wesley, 2001.

[Moy98]: John T. Moy. *OSPF: anatomy of an Internet Routing Protocol*. Massachusetts: Addison-Wesley, 1998.

[Par98]: William R. Parkhurst. *Cisco Router OSPF. Design & Implementation Guide*. McGraw-Hill, 1998.

[Tho02]: Stephen A. Thomas. *IP Switching and Routing Essentials. Understanding RIP, OSPF, BGP, MPLS, CR-LDP, and RSVP-TE*. John Wiley & Sons, Inc., 2002.

[WP02+:Dijkstra]: Autorenkollektiv. *Dijkstra-Algorithmus*. Wikipedia, 2002-2010. Revision 74931166.  
<http://de.wikipedia.org/wiki/Dijkstra-Algorithmus>

[WSR05]: Russ White, Don Slice, Alvaro Retana. *Optimal Routing Design. Techniques for optimizing large-scale IP routing operation and managing network growth*. Indianapolis: Cisco Press, 2005.

## ANHANG

### IOS: Komplette Auszug von „config-router ?“

```
Switch(config)#router ospf 100
Switch(config-router)#?
Router configuration commands:
  area                OSPF area parameters
  auto-cost           Calculate OSPF interface cost according to
                    bandwidth
  bfd                 BFD configuration commands
  capability          Enable specific OSPF feature
  compatible          OSPF compatibility list
  default             Set a command to its defaults
  default-information Control distribution of default information
  default-metric      Set metric of redistributed routes
  discard-route       Enable or disable discard-route installation
  distance            Define an administrative distance
  distribute-list     Filter networks in routing updates
  domain-id           OSPF domain-id
  domain-tag          OSPF domain-tag
  exit               Exit from routing protocol configuration mode
  help               Description of the interactive help system
  ignore             Do not complain about specific event
  ispf               Enable incremental SPF computation
  limit              Limit a specific OSPF feature
  log-adjacency-changes Log changes in adjacency state
  max-lsa            maximum number of LSAs OSPF process will receive
  max-metric         Set maximum metric
  maximum-paths      Forward packets over multiple paths
  neighbor           Specify a neighbor router
  network            Enable routing on an IP network
  no                 Negate a command or set its defaults
  nsf                Non-stop forwarding
  passive-interface  Suppress routing updates on an interface
  process-min-time   Percentage of quantum to be used before releasing
                    CPU
  queue-depth        OSPF Hello/Router process queue depth
  redistribute        Redistribute information from another routing
                    protocol
  router-id          router-id for this OSPF process
  summary-address    Configure IP address summaries
  timers             Adjust routing timers
  traffic-share      How to compute traffic share over alternate paths

Switch(config-router)#
```

## LIZENZTEXT

Copyright (C) 2010 Sebastian Wieseler.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

All trademarks, related works and pictures are property of their respective owners.