REGIONALES RECHENZENTRUM ERLANGEN [RRZE]

Routing im Intra- und Internet

Netzwerkausbildung – Praxis der Datenkommunikation, 31.01.2018

Helmut Wünsch, RRZE

Dieser Vortrag wird aufgezeichnet.

Die ersten beiden Sitzreihen befinden sich im Kameraradius.

Themen

- Routing – allgemein
  - Statisches Routing
  - Dynamisches Routing
  - Routing im Internet
- Routingprotokolle
  - distance vector (Bsp. RIP)
  - link state (Bsp. OSPF)
  - path vector (Bsp. BGP)

Routing – Was ist Routing?

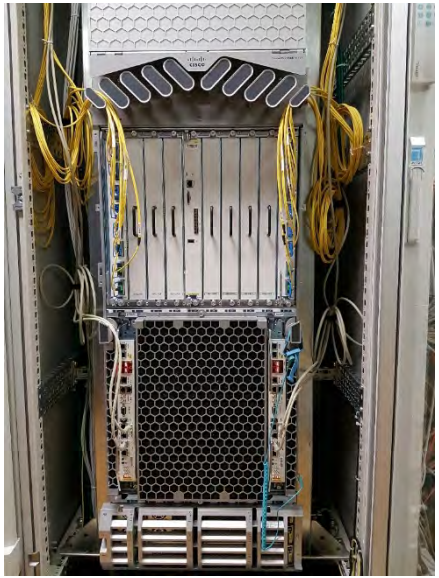
- Alle Rechner innerhalb eines lokalen Netzes (LANs) können direkt miteinander kommunizieren
- Aber: Ein lokales Netz ist (wie der Name schon sagt) lokal begrenzt
- Wollen Rechner eines LANs mit Rechnern eines anderen LANs kommunizieren, braucht es vermittelnde Stationen zwischen den Netzen, sog. *Router*

Routing – Was ist Routing?

- Ein Router ist i.d.R. ein dediziertes Gerät mit mehreren Schnittstellen, an welche
  - jeweils ein lokales Netz (LAN)
  - Oder auch andere, (z.T. weit entfernte) Router angeschlossen sind,
- der Router schaufelt die Pakete zwischen den Schnittstellen hin und her („auf Layer3“)

Beispiele von Routern

Quelle: wikipedia.org



Typischer Router für den Heimbereich:

- → i.d.R. 2 Schnittstellen (“LAN“ <-> “WAN“)
- → Klein, einfach, sparsam
- → 10 bis 100Mbit/s Forwarding

Hochleistungsrouter des DFN im X-WiN

- → Sehr viele Schnittstellen (physikalisch oder virtuell)
- → Groß, komplex, teuer
- → Bis zu 64 Terabit/s Forwarding (modular erweiterbar)

Router als Architekturbaustein für komplexe Netzwerke

Gezielter Einsatz von Routing im eigenen Netzwerk:
Prinzip: Viele kleinere statt wenige große LANs

- Logische (effiziente) Trennung von Subnetzen
- Skalierung: Sehr Dynamisches Wachstum von Netzwerken möglich
- Sicherheit (Möglichkeit der Zugriffskontrolle)
- Stabilität (Forwarding vs. Flooding)
- Erhöhung von Redundanz und Performance

Funktionsweise: IP-Kommunikation innerhalb eins LANs

→ IP-Adressen der Quell- und Zielrechner sind im gleichen LAN (z.B. beide im Netz `192.168.1.*/24`)

- Kein Router nötig: Quellrechner kann dann das IP-Paket direkt an MAC-Adresse des Zielrechners im LAN schicken
- Ermittlung der MAC-Adresse des Zielrechners:
  - Bei IPv4: per ARP-Request
  - Bei IPv6: per NDP-Request
- Woher weiß der Quellrechner, dass die Zieladresse im gleichen LAN angesiedelt ist?
- Durch die Netzmaske des jeweiligen LANs.
  - definiert die „Größe“ bzw. den Adressbereich des IP-Netzes
  - Sollte tunlichst auf jedem Rechner im selben LAN gleich konfiguriert sein!

/slash	# Hosts	Netmask	Wildcard
/30	4	255.255.255.252	0.0.0.3
/29	8	255.255.255.248	0.0.0.7
/28	16	255.255.255.240	0.0.0.15
/27	32	255.255.255.224	0.0.0.31
/26	64	255.255.255.192	0.0.0.63
/25	128	255.255.255.128	0.0.0.127
/24	256	255.255.255.0	0.0.0.255
/23	512	255.255.254.0	0.0.1.255
/22	1,024	255.255.252.0	0.0.3.255
/21	2,048	255.255.248.0	0.0.7.255
/20	4,096	255.255.240.0	0.0.15.255
/19	8,192	255.255.224.0	0.0.32.255
/18	16,384	255.255.192.0	0.0.63.255
/17	32,768	255.255.128.0	0.0.127.255
/16	65,536	255.255.0.0	0.0.255.255
/15	131,072	255.254.0.0	0.1.255.255
/14	262,144	255.252.0.0	0.3.255.255
/13	524,288	255.248.0.0	0.7.255.255
/12	1,048,576	255.240.0.0	0.15.255.255
/11	2,097,152	255.224.0.0	0.31.255.255
/10	4,194,304	255.192.0.0	0.63.255.255
/9	8,388,608	255.128.0.0	0.127.255.255
/8	16,777,216	255.0.0.0	0.255.255.255

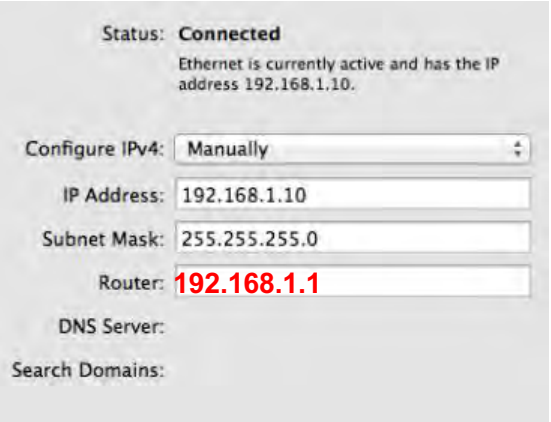
→ Quelle: c128.com

Funktionsweise: IP-Kommunikation zwischen verschied. LANs

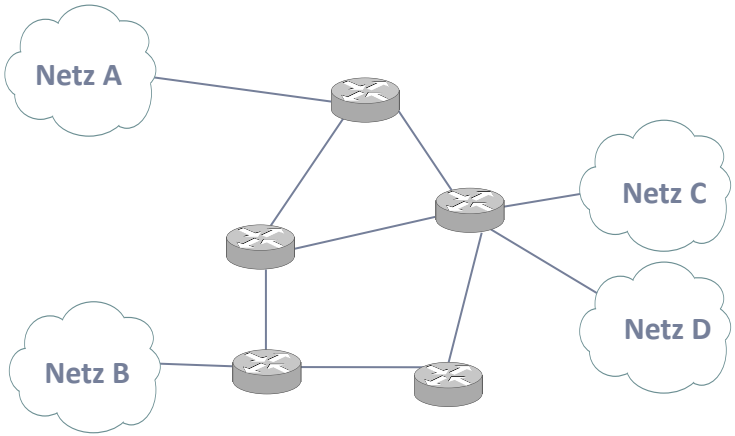
→ IP-Adressen von Quell- und Zielrechner in unterschiedlichen LANs

- D.h. Quellrechner ermittelt mit seiner Netzmaske ermittelt, dass Zielpartner nicht im selben LAN sitzt

  Bsp: Kommunikation `192.168.0.10` nach `10.78.0.35`
- Quellrechner schickt das IP-Paket an MAC-Adresse des Routers
- Dazu muss dem Quellrechner der Router im LAN bekannt sein!
  - › Router wie Netzmaske essentieller Teil der Netzwerkkonfiguration
  - › Auf Endgeräten gerne auch als „default Gateway“ bezeichnet
- Router leitet IP-Paket weiter:
  - › An MAC-Adresse des Empfängers im Ziel-LAN, sofern er dieses direkt erreichen kann
  - › An anderen Router



Routingtabelle



- Woher weiß der Router, wo welche Zielnetze liegen?
  - Bei Netzwerken mit mehr als einem Router keine triviale Frage mehr
  - Jeder Router muss eine jederzeit gültige „Adressdatenbank“ führen, die sog. Routing-Tabelle.

Routingtabelle

Ziel	Intf.	Metrik
131.188.78.0/23	Interface P	1
192.168.0.0/24	Interface A	1
192.168.0.0/24	Interface B	2
10.78.0.0/16	Interface C	1
0.0.0.0/0 (“default route“)	Interface D	10

→ Exemplarische Routingtabelle

Routing-Tabelle enthält Infos zu

- andere LANs am Router („directly connected“),
- Entfernte LANs an anderen Routern erreichbare Netze („next hop“ routing)

- Aufbau einer Routingtabelle, Mindestinfo:
  - Zielnetz
  - Zielinterface
  - Metrik
- Routenauswahl:
  - Longest Prefix Match
  - Metrik
- Größe der Routingtabelle: 2 (DSL-Router)

  bis ~450.000 (Internet BGP Router)

Erstellen der Routingtabellen

Wie wird die Routingtabelle aufgebaut?

Statisches Routing:

- Manuelle Konfiguration der Routing-Tabelle auf jeweiligem Router

Dynamisches Routing:

- Alle Router im Netzwerk unterhalten sich untereinander und bauen Routingtabelle selbstständig über Routingprotokolle auf

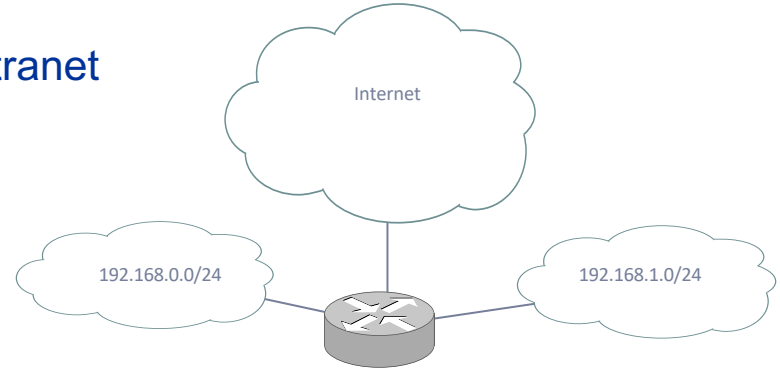
Statisches Routing (I)

Einfachstes Bsp.: DSL-Router zuhause

- z.B. „Fritzbox“
- Routet i.d.R. nur zwischen zwei Netzen: Lokales Heimnetz und Internet
- Triviale Routingtabelle:
  - Heimnetzwerk (z.B. 192.168.178.0/24)→ LAN-Port 1-4
  - Internet(„*default route*“) → LAN-Port 5

Anderes Bsp.: Firewall-Router für kleines Firmen-Intranet

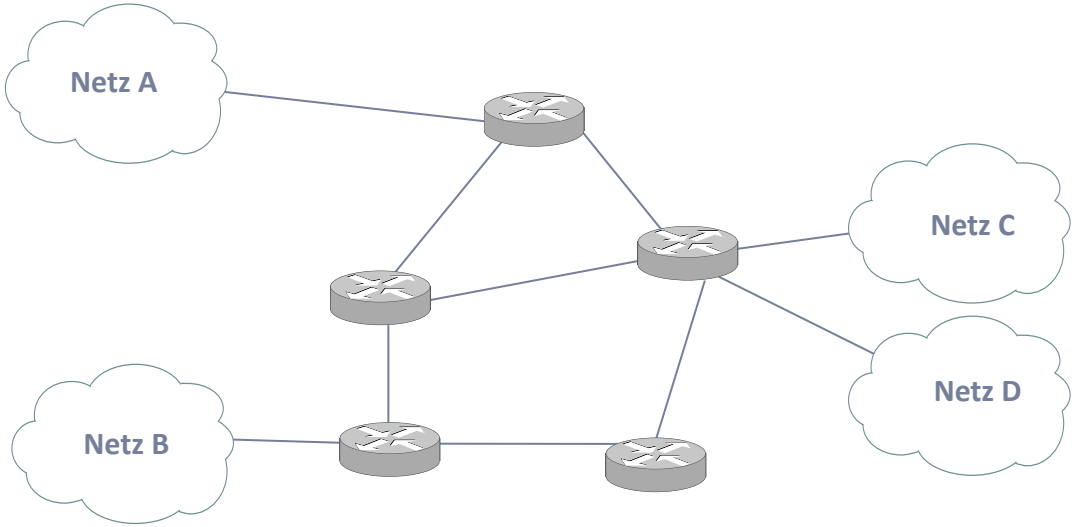
- Z.B. jeweils eine Netzwerkschnittstelle für
  - Bürorechner 192.168.0.0/24
  - Servernetz 192.168.1.0/24
  - Internet
- zugehörige statische Routingtabelle:



Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
0.0.0.0	0.0.0.0	0.0.0.0	U	0	0	0	ppp0

Statisches Routing (II)

Grenzen von statischen Routing: Mehrere vermaschte Router



Hoher Aufwand bei Anlegen, Löschen, Umzug, … von Netzen:

→ Routinginformationen müssen auf jeden Router manuell nachgetragen werden

→ Bei mehreren vermaschten Routern wird statisches Routing sehr schnell unhandhabbar!

Dynamisches Routing: Sinn und Zweck

Durch dynamisches Routing sollen Router…

- Routinginformationen selbständig untereinander austauschen
- selbständig die Netztopologie „lernen“
- somit selbständig für jedes Paket den jeweils besten Weg zum Ziel ermitteln
- selbständig auf Veränderungen in der Topologie reagieren
- gut wie möglich Fehler vermeiden (z.B. Schleifentopologien)

Statisches vs. Dynamisches Routing

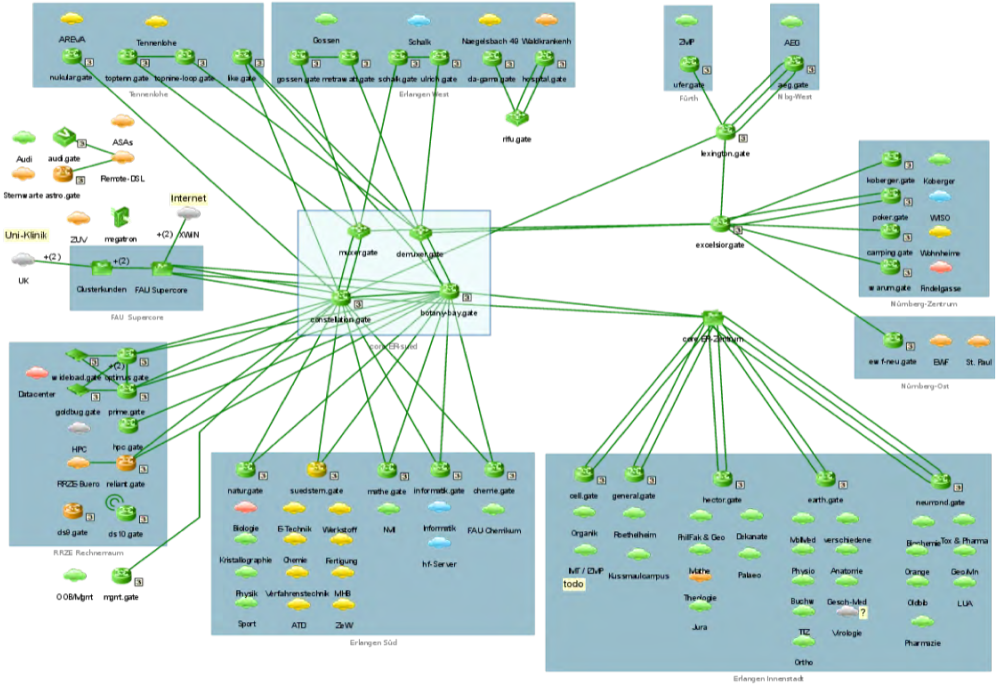
Gegenüberstellung

- Dynamisches Routing kann durch falsche Informationen gestört werden
- Dynamisches Routing erzeugt Netzlast
- Statisches Routing nur bei einfachen Netztopologien handhabbar
- Keine Backup-Pfade bei statischem Routing
- Statisches Routing mit mehr als einem Router ist arbeitsintensiv bei Änderungen und fehleranfällig

Bsp. Dynamisches Routing (I)

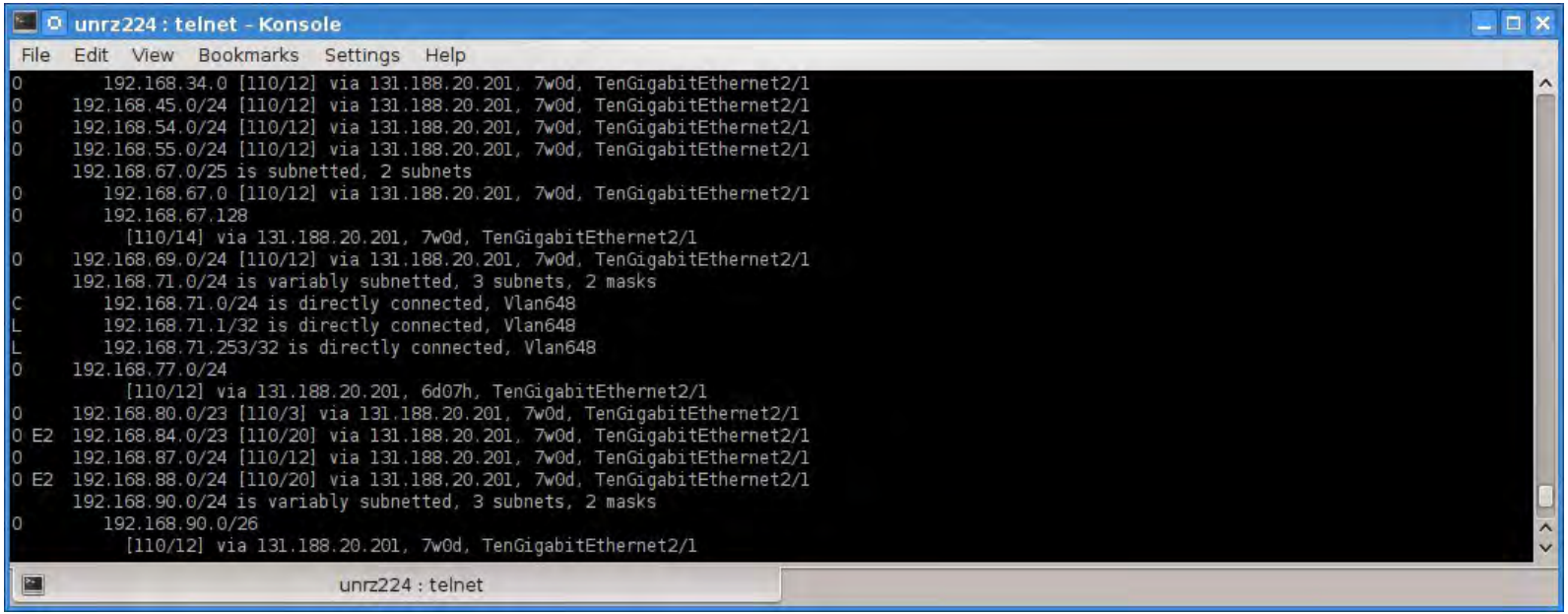
FAU Backbone-Netz

- Universitärer Backbone aufgebaut mit ca. 30 Backbone-Routern, verteilt auf mehrere Städte
- Statisches Routing auf jeden Router indiskutabel
- Änderung des Netzes an beliebiger Stelle lässt Backbone automatisch reorganisieren
- verschiedene Subnetze in den Routingtabellen!



Bsp. Dynamisches Routing (II)

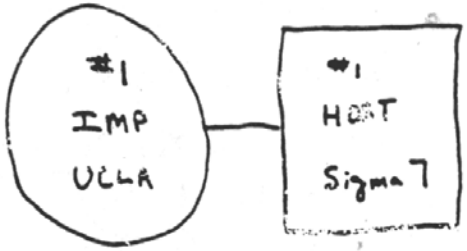
Uni-Backbone: Auszug Routingtabelle



Bsp. Dynamisches Routing (III)

Das Internet

- 1969 als „ARPANET“ entstanden
- Durch die ARPA als Forschungsprojekt initiiert
- Gründungsmythos: Kommunikationsnetz, robust gegen „nuklearen Zerstörung“
- Tatsächlich: Projekt, um einzelne Uni- und Forschungsnetze im Land dezentral und effizient (über Telefonleitungen) zusammenzuschalten
- Grundprinzip geblieben bis heute: Internet nach wie vor aufgebaut aus einzelnen, unabhängig voneinander verwalteten Netzwerken von Provider/Uni/Regierung,…
- Einzelnetzwerke im Internet auch bezeichnet als „AS“ (Autonome Systeme)



THE ARPA NETWORK

SEPT. 1969

1 NODE

FIGURE 6.1 Drawing of September 1969 (Courtesy of Alex McKenzie)

Historik / Internet „Arpanet“ im Frühstadium

September 1969:
1 Knoten

Uni Kalifornien

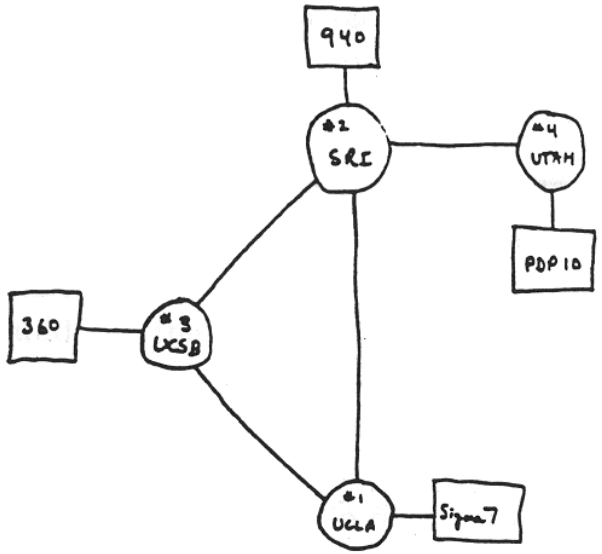


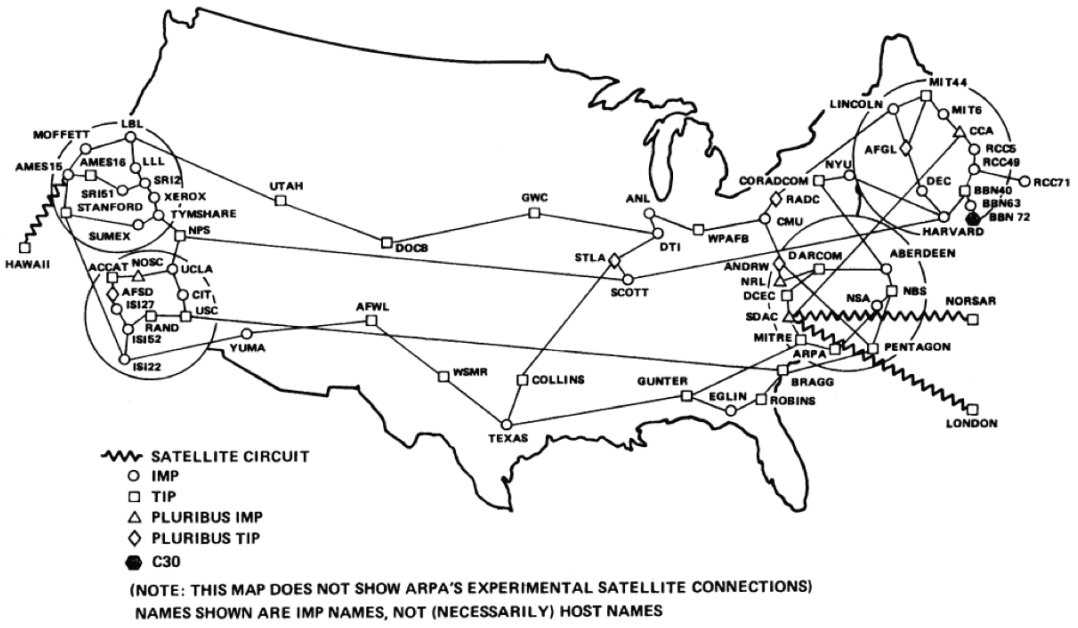
FIGURE 6.2 Drawing of 4 Node Network (Courtesy of Alex McKenzie)

Historik / Internet „Arpanet“ im Frühstadium (2)

Dezember 1969:
4 Knotenpunkte

- Kalifornien
- Utah
- Stanford
- Santa Barbara

Historik / Internet „Arpanet“ im Frühstadium (3)



Historik

Weiterer Werdegang:

- Abspaltung des „Milnet“ aus dem Arpnet
- Arpnet -> NSFNet (Abschaltung Arpnet 1989)
- NSFNet -> Internet (90er)

Routing im Internet

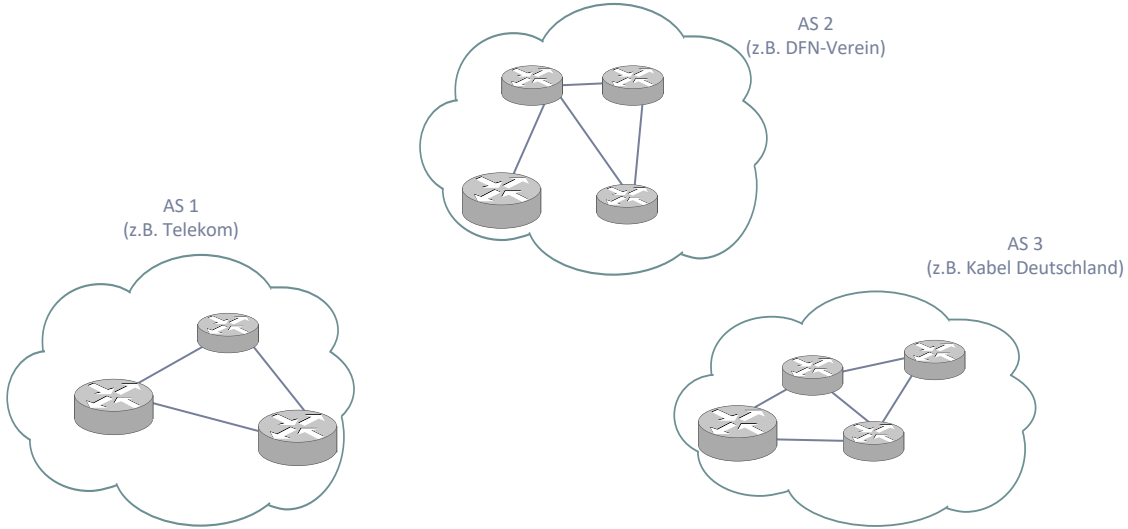
Heutiges Internet

- Bis heute besteht das Internet aus zigtausend Einzelnetzwerken, jeweils unter Kontrolle ihres Betreibers (Telcos, Firmen, Unis, Behörden,…)
- Diese Einzelnetzwerke werden auch bezeichnet als „AS“ (Autonome Systeme)
- AS sind somit die „Einzelbausteine“ des Internets

Vom AS zum Internet (I)

„Autonomes System“:

- IP-Netzwerk unter organisatorisch eigener Verwaltung
- I.d.R. leistungsfähige Netze Privater oder Öffentlicher Betreiber (Firmen, Unis, Telcos, Behörden,…)



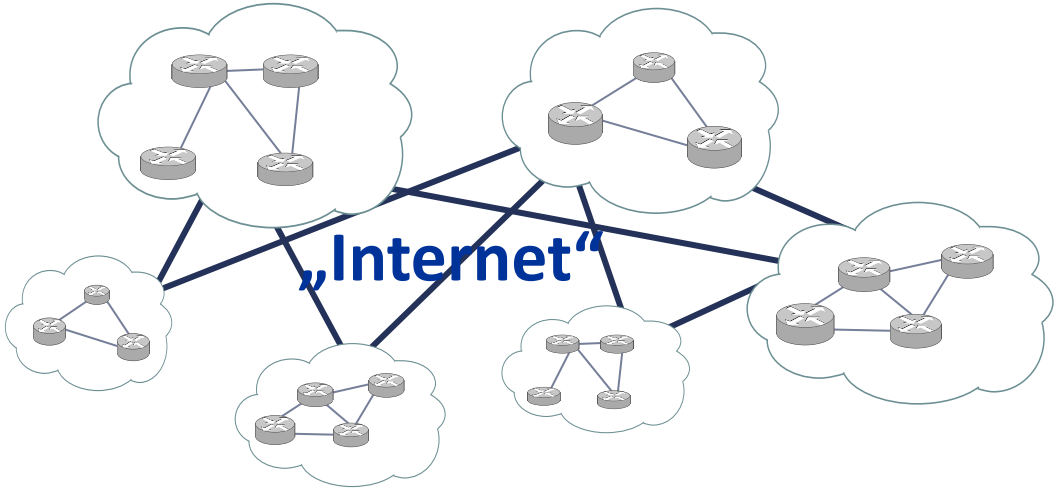
Vom AS zum Internet (II)

Vernetzungen von mehreren AS

Wenn sich nun AS untereinander vernetzen wollen…

- …schalten sie in eigener Verantwortung Verbindungen zwischen sich…

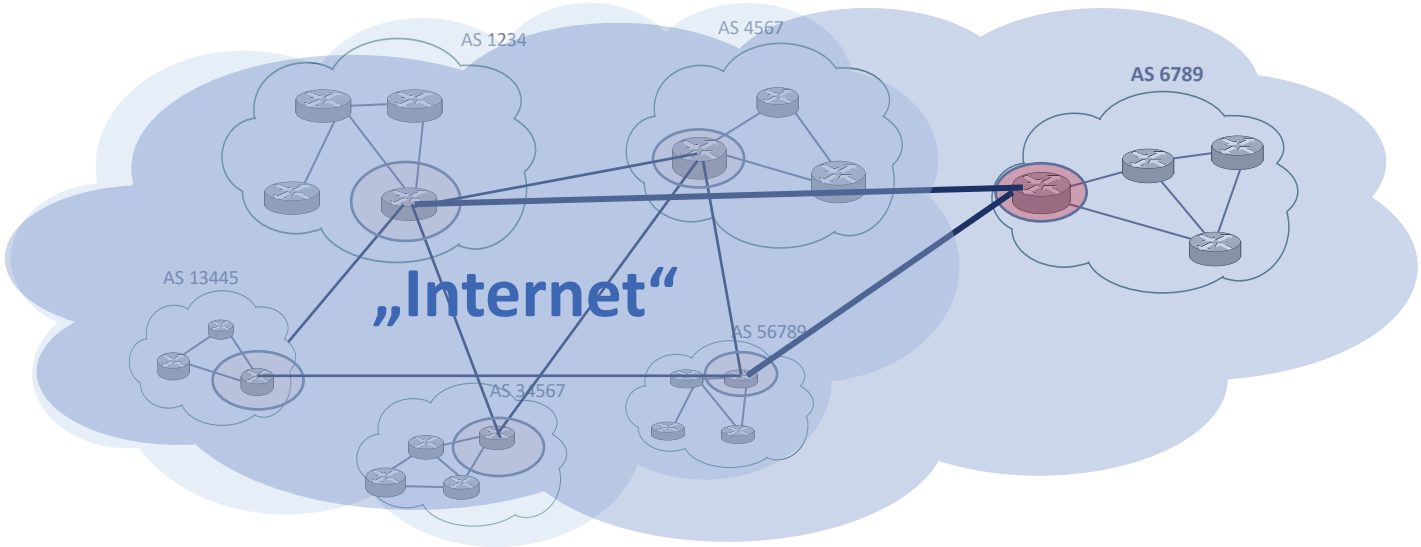
  („Peering“ bzw. „Transit“ Verhandlungen)
- …und bilden dadurch zusammen ein immer größer werdendes Netz der Netze, das **“Inter-Net”**



Vom AS zum Internet (III)

Für die Teilnahme eines neuen AS am Internet braucht es in der Praxis…

- eine öffentliche AS-Nummer vom RIPE (Bsp.: T-COM (AS3320), DFN (AS680), MNet/Nefkom (AS8767))
- Mind. einen dedizierten sog. *Border-Router* im AS als Verbindungsschnittstelle nach außen
- Peering/Routingvereinbarungen zu Border-Routern mind.(!) zwei anderer AS
- Konfiguration und Aktivierung des sog. „Border Gateway Protocol“ (BGP) auf dem Border-Router (dynamisches Routingprotokoll)
- → Alle Router synchronisieren ihre Routinginformationen: „Das Internet wächst“

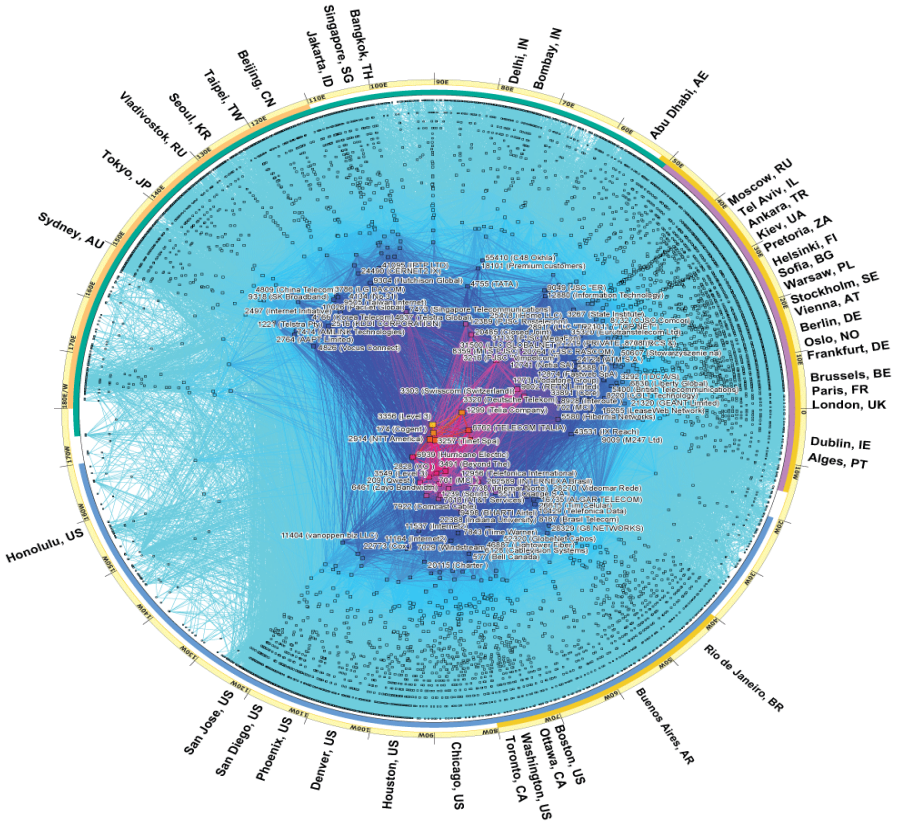


Vom AS zum Internet (IV)

Nach Einbindung eines AS in das Internet:

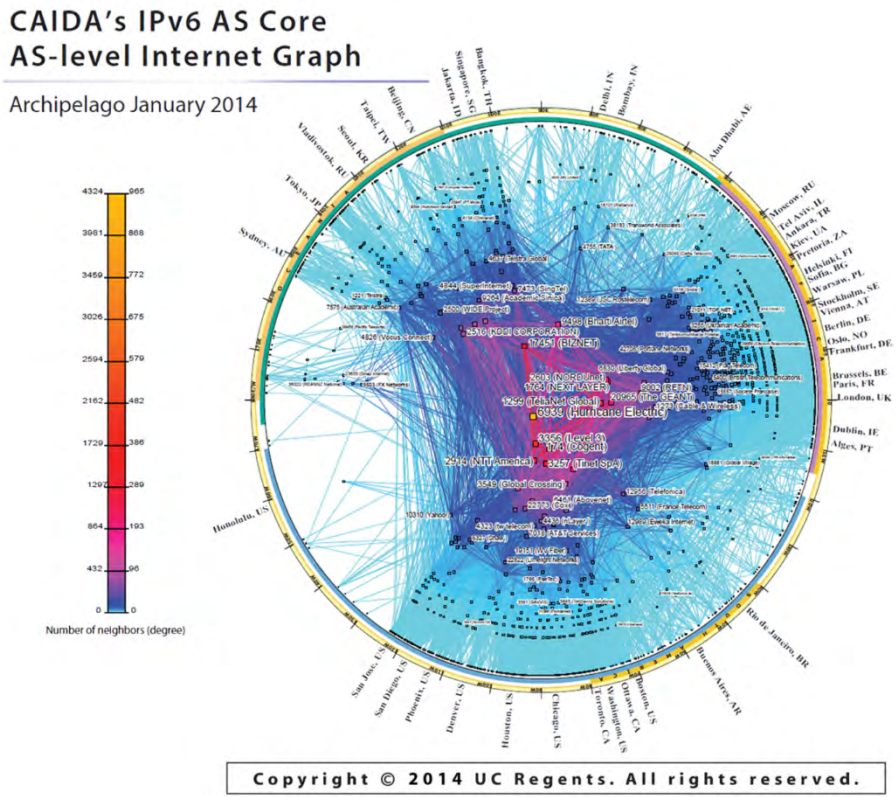
- Jeder Border-Router synchronisiert sich mit seinen Partnern per BGP (Border Gateway Protokoll):
- Jedes AS beheimatet einerseits nur einen Bruchteil aller IPv4/v6-Netze
  - (Bsp. AS-680 des DFN: „Heimat“ der meisten Uni-Netze in Deutschland)
- Aber: Jeder Border-Router eines jeden AS kennt die Netze inkl. Routen _aller_ anderen AS
- → Extrem große Routingtabellen auf allen Border-Routern:
  - Derzeit (2015): > 450.000 IPv4-Routen müssen auf jedem Border-Router eines AS vorgehalten werden

Internet heute: Vermaschung per IPv4



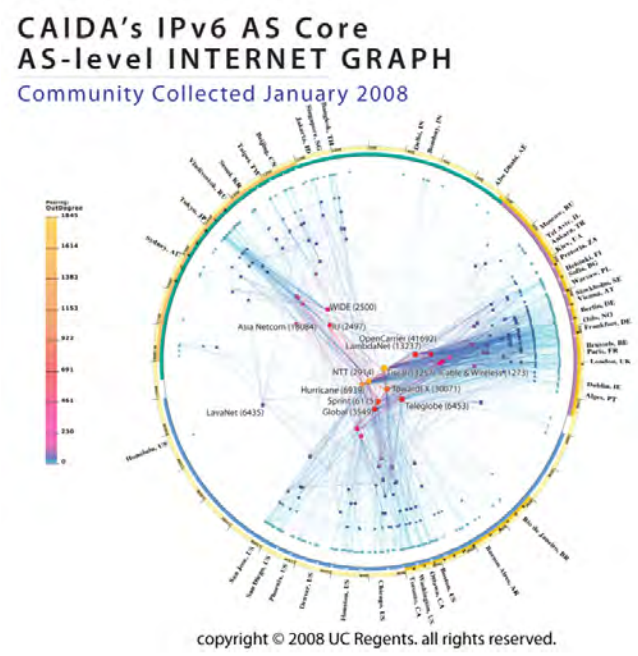
Quelle: www.caida.org
Stand: Feb 2017

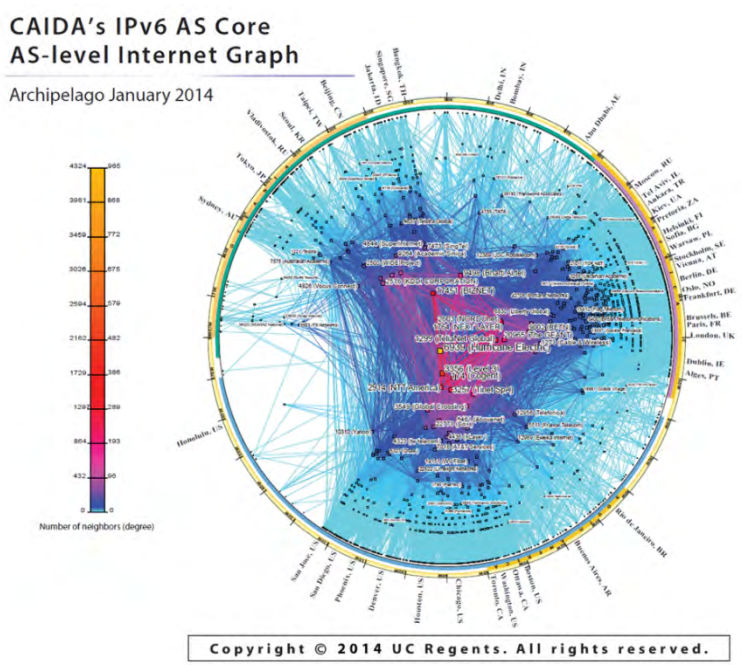
Internet heute: Vermaschung per IPv6



Quelle: www.caida.org

„Erwachsenwerden“ von IPv6 2008 vs. 2014





Quelle: www.caida.org

Vom AS zum Internet (V)

Autonome Systeme als „anarchische“ Strukturen

allein die

rnet

n

n/

- AS-Be odex
- Denno

DYNAMISCHE ROUTINGPROTOKOLLE

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

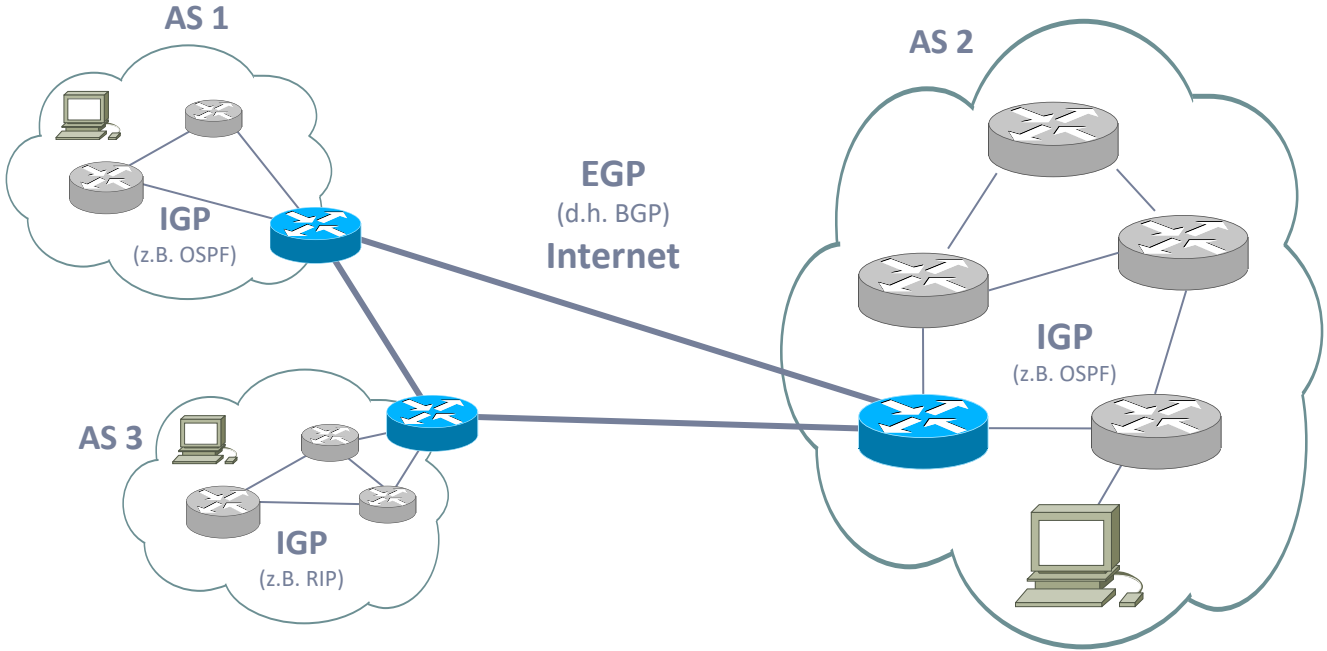
Dynamische Routingprotokolle

Logische Unterscheidung

- Internal Gateway Protocol (IGP)
  - → Für dynamisches Routing innerhalb eines AS (z.B. Universitätscampus)
    - Z.B. RIP, OSPF, IGRP, …
- External Gateway Protocol (EGP)
  - → Für dynamisches Routing zwischen verschiedenen AS (→ „Internet“)
    - Einziger praktischer Vertreter: BGP: Border Gateway Protocol

IGP vs. EGP

Zusammenspiel von IGPs und EGP im Internet



Wie funktionieren Routingprotokolle?

Technische Unterscheidung dynamischer Routingprotokolle

- distance vector
  - z.B. RIP (IGP)
- link state
  - z.B. OSPF (IGP)

Internal Gateway Protocol (IGP)

- path vector
  - z.B. BGP (EGP)

External Gateway Protocol (EGP)

Distance-Vector-Protokolle

Grobe Funktionsweise:

- jeder Router pflegt Tabelle mit gelernten Pfaden zu versch. Zielnetzen
- periodische Weitergabe (i.d.R. 30sek) dieser Tabelle jeweils an Nachbar-Router
- Nachbar-Router updaten ggf. mit diesen Daten ihre Tabellen und senden Ihrerseits beim nächsten Update ihre Tabelle an Nachbarn
- Änderungen „sprechen sich langsam im Netz rum“
- „Distanz“ als einzige Berechnungs-/Bewertungsgrundlage (Metrik) bei mehren Routen zum gleichen Ziel

Problem: langsame Konvergenz bei Routingänderungen

Distance-Vector-Protokolle: RIP (I)

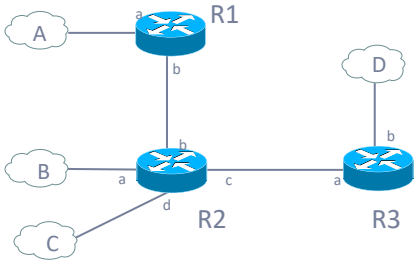
Bekanntester Vertreter: RIP

Eigenschaften

- RIP - Routing Information Protocol
- IGP-Einsatz, d.h. nur innerhalb von AS bzw. Intranets verwendet
- entwickelt von Ford und Fulkerson, daher auch Ford-Fulkerson Algorithmus
- definiert in RFC 1058, viele Erweiterungen
- relativ einfaches Distance-Vector basiertes Protokoll
- „Hop-Count“ als einzige Metrik, d.h. keine explizite Angabe von Pfadkosten möglich

Distance-Vector-Protokolle: RIP (II)

RIP, Grobes Prinzip

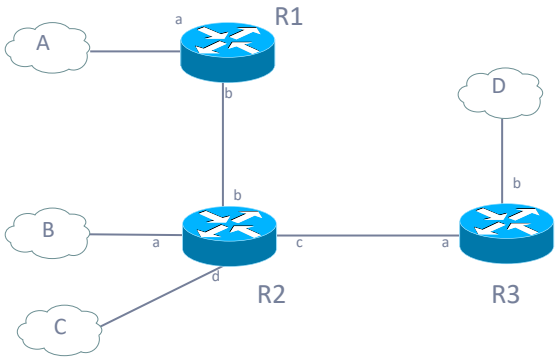


- Jeder Router besitzt Routingtabelle mit „Hopcount“ als Bewertungskriterium(Metrik)
  (Hopcount == Anzahl der Router, über die das Ziel erreicht werden kann)
- Router schicken ihre Tabellen alle 30 sek. an ihre Nachbarn
- Jeder Router verbessert ggf. mit den Infos der Nachbarn seine Routingtabelle
- Timeout-Mechanismus:
  - nach 180 Sek. ohne neues Update: Gelernte Route wird als unbrauchbar markiert
  - nach 240 Sek. ohne Update: Gelernte Route wird gelöscht

Routingtabelle von R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

Distance-Vector-Protokolle: RIP, Beispiel



Ausgangsfall: Router kennen nur ihre direkten Nachbarn

R1

Ziel	Intf.	Hops
A	a	1
R2	b	1

R2

Ziel	Intf.	Hops
B	a	1
C	d	1
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
D	b	1
R2	a	1

1. Update: Nachbarrouter tauschen ihre Routingtabellen aus
→ R2 lernt dabei Routen zum Ziel A und D, R1 und R3 wiederum zu den Zielen B,C,R3 bzw. B,C,R1

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
R2	b	1
R3	b	2

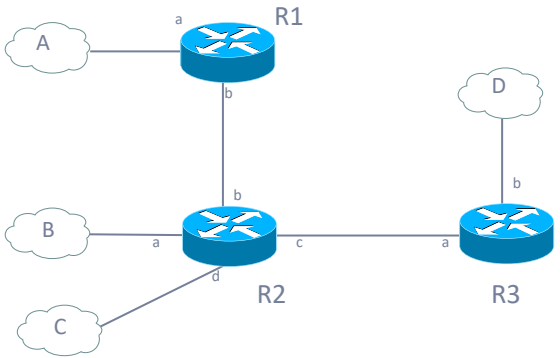
R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

Distance-Vector-Protokolle: RIP, Beispiel



Ausgangspunkt nach 1. Update

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
R2	b	1
R3	b	2

R2

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

2. Update: Nachbarrouter tauschen wieder ihre Routingtabellen aus und updaten die Ihrige. Für R2 ergeben sich keine neuen Informationen, er hält seinen Zustand stabil. R1 und R3 lernen von R2 noch das Netz D bzw. A.

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

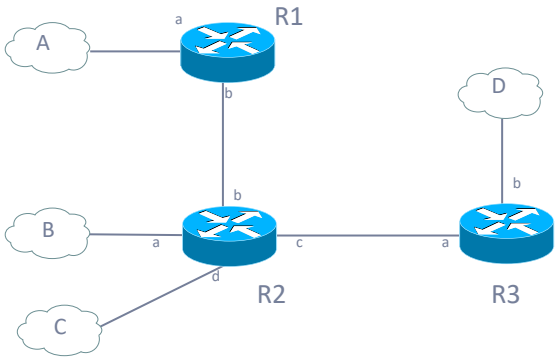
R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

Distance-Vector-Protokolle: RIP, Beispiel



Ausgangspunkt nach 2. Update:

R1

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

3. Update: Ein erneuter Austausch der Routingtabellen bringt keinem Router mehr eine Änderung.
→ Nach diesem Durchgang sind die Routingtabellen auf alle Routern stabil

R1 [stabil]

Ziel	Intf.	Hops
A	a	1
B	b	2
C	b	2
D	b	3
R2	b	1
R3	b	2

R2 [stabil]

Ziel	Intf.	Hops
A	b	2
B	a	1
C	d	1
D	c	2
R1	b	1
R3	c	1

R3 [stabil]

Ziel	Intf.	Hops
A	a	3
B	a	2
C	a	2
D	b	1
R1	a	2
R2	a	1

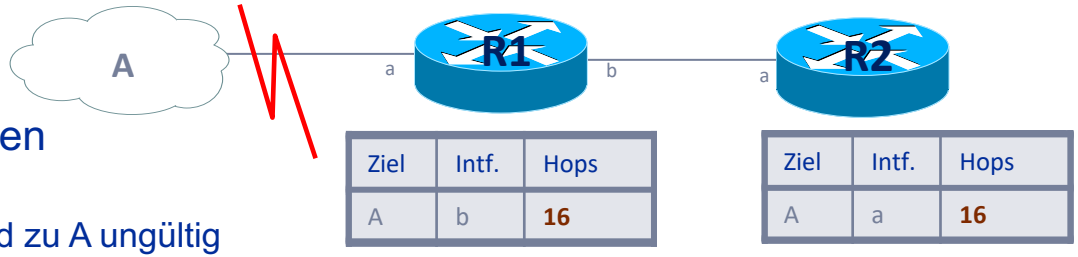
RIP: Probleme: Wenn der Bagger…

Routingschleifen: „Counting to Infinity“

Szenario:

- Router R1 hat Route zu Netz A mit Distanz=1 (direkt angeschlossen)
- Router R2 hat ebenfalls Route zu Netz A mit Distanz=2 (d.h. via R1)
- Router haben stabilen Tabellenzustand, schicken sich Updates alle 30sek

Fehlerfall auf R1-Seite: Verbindung auf Interface a zu Netz A geht verloren



Ziel	Intf.	Hops
A	b	16

Ziel	Intf.	Hops
A	a	16

- R1 aktualisiert seine Routingtabelle: Pfad zu A ungültig (Hopcount 1 auf 16 gesetzt)
  (Schafft es R1 sein Routing-Update vor R2 zu verschicken, ist das auch kein Problem)
- Aber: Falls R2 **vor** R1 sein Routing-Update verschickt, wird R1 eine neue Route für A von R2 lernen (mit Distanz =3)
- Diese Route wird R1 wieder an R2 verbreiten, R2 aktualisiert dann seine Route für A mit Distanz=4 usw. …
- Erst bei Erreichen von Hopcount 16 (nach ca. 7min!) ist die Route zu A endlich ungültig gelöscht

RIP: Erweiterung

RIP, Ansätze zur Problemlösung

- Split Horizon
  - Grundidee: es macht keinen Sinn, Routen in die Richtung weiterzugeben, aus der man sie bekommen hat
- Split Horizon with Poison Reverse
  - Grundidee: sende Routinginformationen in die Richtung, aus der sie gekommen sind mit Metrik 16 (unendlich) zurück
- Holddown
  - Grundidee: akzeptiere keine Routinginformationen zu einem Ziel, für welches man selbst eben Informationen verbreitet hat, für eine gewisse Zeit

RIP: Nachteile

Hauptnachteile von RIP bzw. Distanzvektorprotokollen

- Relativ langsame Konvergenz (Minutenbereich)
- Nur HopCount als Metrik
- Nur für kleinere Netze geeignet

→ Für größere lokale Netze mit erweiterten Anforderungen: Link-State-Protokolle

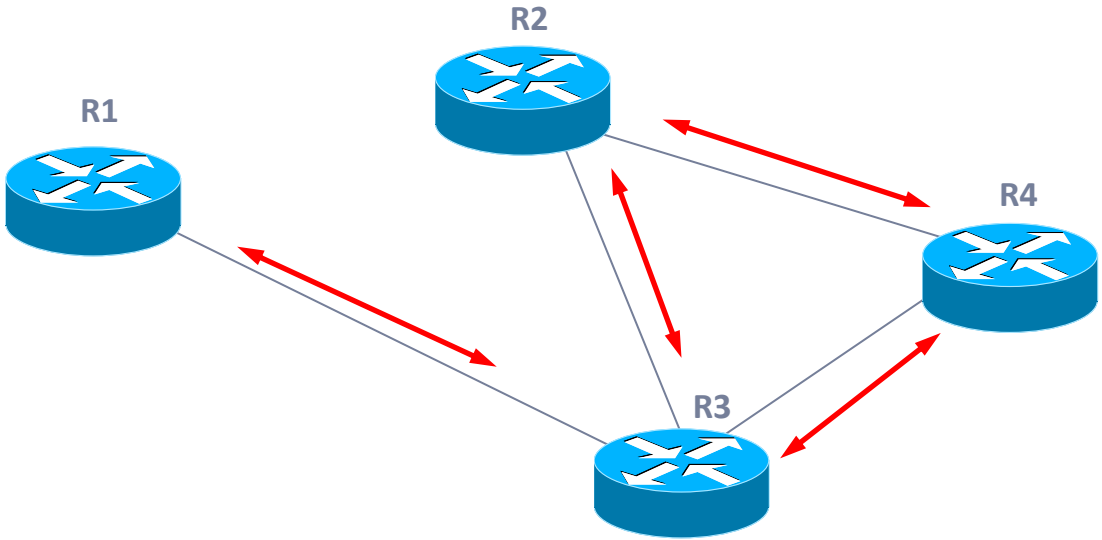
Link-State-Protokolle: Prinzip

- Benachbarte Router bauen eine permanente Verbindung („Adjazenz“) auf und tauschen asynchron Nachrichten über Veränderungen des Netzes aus
- Änderungen auf einem Router werden unmittelbar (ggf. inkrementell) an anderen Router gesendet (→ sehr schnelle Reaktion)
- Nach Konvergenz hat jeder Router eine komplette Sicht auf das gesamte Netz (als vermaschter Graph)
- Jeder Router berechnet dann auf Basis dieser Sicht eine optimale Routingtabelle (Graphentheorie, Spannbaumprinzip)

Link-State-Protokolle: Arbeitsweise (1)

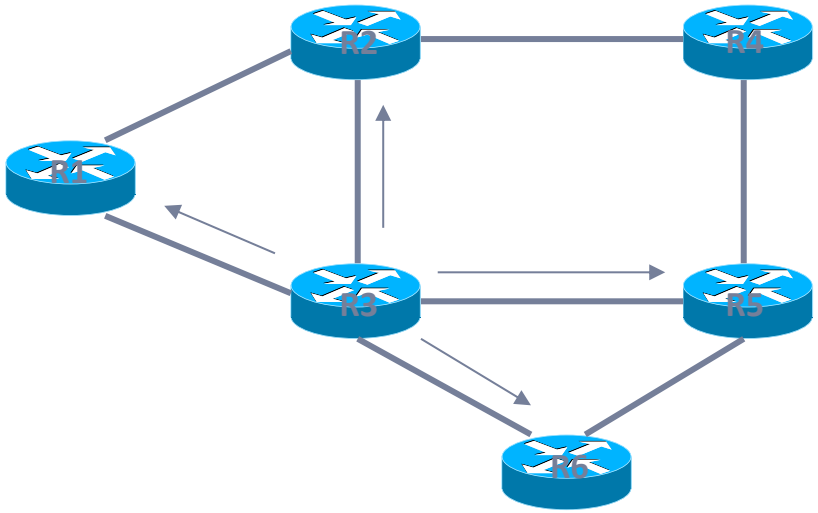
„Link-State“

Jeder Router hält permanent eine Verbindung zu seinen unmittelbaren Nachbarn offen. Darüber findet ein asynchroner Nachrichtenaustausch statt, über sog. LSAs: (Link State Advertisements)



Link-State-Protokolle: Arbeitsweise (2)

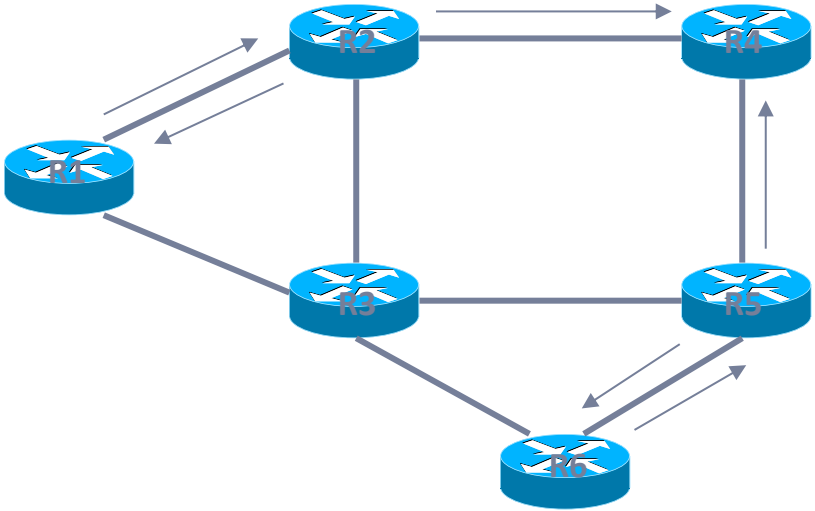
Prozedere beim Start eines Routers: „Flooding“



Bsp.: Router R3 fährt hoch.
Er sendet als Erstes seine Routinginformationen (d.h. lokal angeschlossene Netze) an alle seine Nachbarn.

Link-State-Protokolle: Arbeitsweise (3)

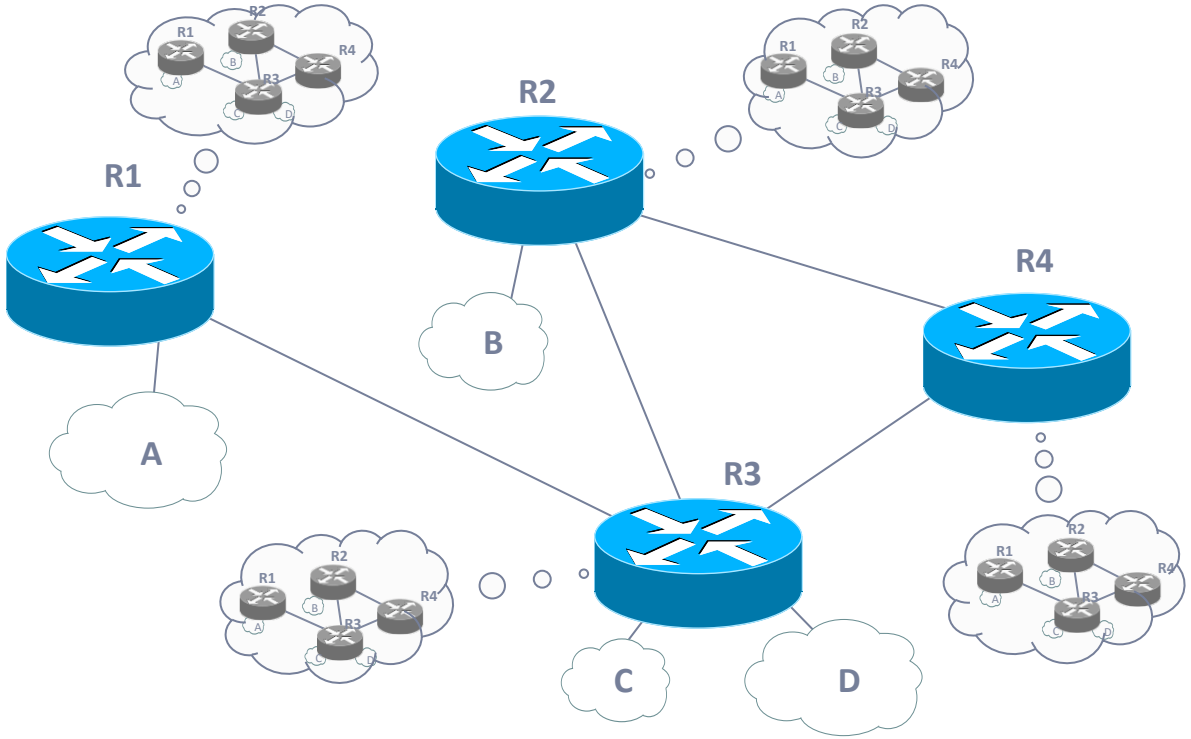
Weitergabe der Routinginformationen durch das Netz



Die Nachbarn aktualisieren augenblicklich ihre Datenbasis und schicken daraufhin Updates an ihre Nachbarn: Die Information wird „flutend“ unmittelbar durch das gesamte Netz durchgereicht (sehr schnell!)

Link-State-Protokolle: Arbeitsweise (4)

Nach kurzer Zeit hält jeder Router danach eine stets aktuelle Sicht („Graph“) des kompletten Netzwerks vorrätig



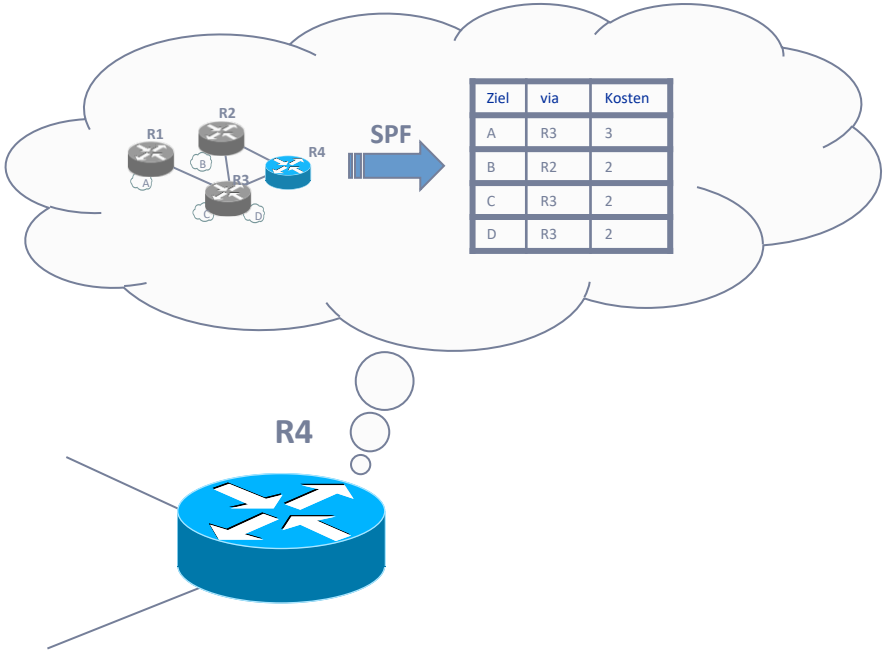
Link-State-Protokolle: SPF (1)

Danach: Jeder Router berechnet sich aus Graph seine Routingtabelle

- Grundlage: Sog. „Shortest Path“ Algorithmen
- Findet kürzeste Wege in vermaschten Graphen von ggf. Start und Zielpunkt
- Jeder Router berechnet so seine optimale Routingtabelle zu allen Zielen
- Bei Topologie-Veränderungen: Flooding der Änderung und Neuberechnung
- Konvergenzgeschwindigkeit von nur wenigen Sekunden
- bekanntester SPF-Algorithmus: Dijkstra

Link-State-Protokolle: SPF (2)

Berechnung der Routingtabelle individuell auf jeden Router mittels SPF-Algorithmus (inkl. Kostenmaß!)



Link-State-Protokoll: OSPF(1)

- Bekanntester Vertreter: OSPF – Open Shortest Path First
- Das am meisten verbreitete Link-State-Routingprotokoll
- entwickelt von J. Moy
- OSPFv1, v2 oder v3 (IPv6)
- IGP-Einsatz, d.h. nur innerhalb von AS verwendet

Link-State-Protokoll: OSPF(2)

- Vorteile
  - Kryptographisch abgesichert (MD5 Checksum)
  - flexible Metriken für Routingentscheidungen
    (z.B. Distance, Hop-Count, $$)
  - Wertebereich für Metrik von 1 bis 65535
  - Skalierbarkeit
  - Unterstützung von mehreren Wegen
    (equal path load sharing, unequal path load sharing)

Pfad-Vektor-Protokolle

Situation Internet

- Für Routing zwischen Autonomen Systemen (→ Internet) sind sowohl Link-State als auch Distanz-Vektor-Protokolle nicht geeignet:
  - Link-State-Protokolle wären im Internet praktisch nicht einsetzbar („Flutorgie“)
  - Distanz-Vektor-Protokolle dagegen weisen bei größeren Netzen zu gravierende Nachteile auf (z.B. Hang zur Schleifenbildung)

→ Entwurf eines neuen Protokolls fürs „Inter-AS-Routing“ (d.h. Internet): Pfad-Vektor-Protokolle

Pfad-Vektor-Protokolle (II)

Bsp. BGP (I)

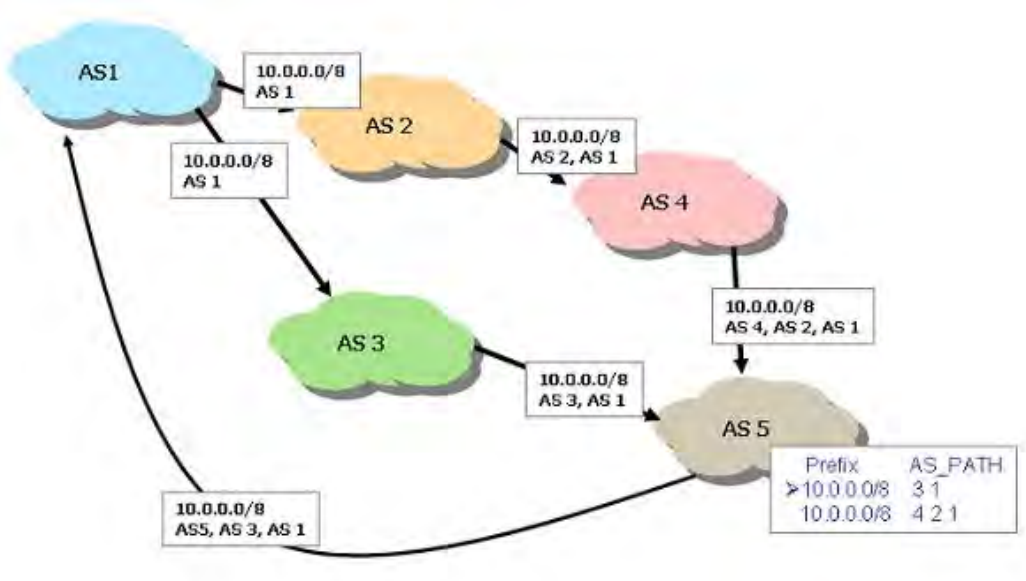
- Prinzip wie Distanz-Vektor Protokoll (vgl. RIP)
  - Erinnere: Distanz-Vektor: Routing-Update enthält Ziel und als Metrik zugehöriger Hopcount
  - Pfad-Vektor: Routing-Update enthält Ziel und als Metrik Pfad der bereits durchlaufenen Router (genauer: AS-Nummern) in Form einer Liste
- Vermeidung von Routing-Schleifen
  - Updates werden verworfen, sobald eigener Router in Pfadliste eines Updates auftaucht.
- Einziger praktischer Vertreter: BGP

Ziel	Intf.	Metrik (Hops)
A	a	**3**
B	b	**4**

Ziel	Intf.	Metrik (Pfad)
A	a	**„R2-R3“**
B	b	**„R2-R3-R5“**

Pfad-Vektor-Protokolle (III)

Bsp.: Prinzip BGP Routing-Update (ausgehend von AS1)



Quelle: http://routemyworld.com/wp-content/uploads/2008/12/bgpas-path.jpg

BGP (III)

Prinzipielle Nachteile von BGP:

- (Keine Möglichkeit der Lastverteilung)

  nur eine Route pro Netz wird ausgewählt
- Auswahl nur nach Anzahl AS, nicht jedoch nach Hops

  Anzahl der Hops innerhalb eines AS unklar!
- keine Berücksichtigung der Link-Geschwindigkeiten
- Sicherheitsaspekte (Spoofing)
- Problem von *Route-Flaps* und *Update-Bursts*

ORGANISATORISCHES

- Die Vorträge im Überblick
- Andere Vortragsreihen des RRZE
- Ablageorte Vortragsfolien
- RRZE-Veranstaltungskalender / Mailingliste abonnieren
- Themenvorschläge und Anregungen

Weitere Vorträge im Rahmen der „Netzwerkausbildung“

Immer mittwochs (ab 14 c.t.) in Raum 2.049 am RRZE

18.10.2017 – Modelle, Begriffe, Mechanismen
25.10.2017 – Lokale Netze: Switching, Routing, Strukturierung
08.11.2017 – Troubleshooting von WLAN- und VPN-Problemen
15.11.2017 – TCP-/IP-Troubleshooting
29.11.2017 – Handeln mit Adressen – ARP, DHCP, DNS
06.12.2017 – IP-FAU-6 (Teil 1)
13.12.2017 – IP-FAU-6 (Teil 2)
10.01.2018 – Elementare Sicherheitsmaßnahmen: Firewall und Netzzugriff
17.01.2018 – Anschluss von Wohnheimnetzen
24.01.2018 – Traffic Engineering: Proxy, NAT
31.01.2018 – Routingprotokolle
07.02.2018 – E-Mail-Grundlagen

Andere Vortragsreihen des RRZE

Campustreffen „IT-Dienste des RRZE und der FAU”

- immer donnerstags ab 15 Uhr c.t.
- vermittelt Informationen zu den Dienstleistungen des RRZE
- befasst sich mit neuer Hard- & Software, Update-Verfahren sowie Lizenzfragen
- ermöglicht den Erfahrungsaustausch mit Spezialisten

Systemausbildung „Grundlagen und Aspekte von Betriebssystemen und System-nahen Diensten”

- immer mittwochs ab 14 Uhr c.t. (in den Sommersemestern)
- Ergänzung zur Netzwerkausbildung “Praxis der Datenkommunikation”
- führt in den grundsätzlichen Aufbau eines Systems sowie eingesetzte Techniken und Komponenten ein
- richtet sich primär an alle Interessierten (Studierende & Beschäftigte)

Vortragsfolien und Vortragsaufzeichnung

Die Vortragsfolien werden nach der Veranstaltung auf der Webseite des RRZE abgelegt:
www.rrze.fau.de/ausbildung-schulung/veranstaltungsreihen/netzwerkausbildung/

Die meisten Vorträge des RRZE werden aufgezeichnet und können nach der Veranstaltung vom Videoportal der FAU heruntergeladen werden:
www.fau.tv

RRZE-Veranstaltungskalender und Mailinglisten

- Kalender abonnieren oder bookmarken
  - Alle Infos hierzu stehen auf der Webseite des RRZE unter:
    www.rrze.fau.de/veranstaltungen/veranstaltungskalender/

- Mailingliste abonnieren
  - Wöchentliche Terminhinweise werden zusätzlich an die Mailingliste RRZE-Aktuelles gesendet.
  - Auch diese Liste kann man abonnieren:
    https://lists.fau.de/mailman/listinfo/rrze-aktuelles

Themenvorschläge und Anregungen

Themenvorschläge und Anregungen nehmen wir gerne entgegen!

Bitte schreiben Sie uns einfach eine E-Mail an:
rrze-zentrale@fau.de (Betreff: Netzwerkausbildung)

Vielen Dank für Ihre Aufmerksamkeit!

Regionales RechenZentrum Erlangen [RRZE]
Martensstraße 1, 91058 Erlangen
http://www.rrze.fau.de