HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

SLAC 2008

Systemnahes Monitoring: Wie, wo, was?

- Stefan Semmelroggen
- Heinlein Professional Linux Support GmbH
- http://www.heinlein-support.de

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Agenda

- top – der Klassiker
- Speichermanagement
- lsof
- strace
- iostat & iotop
- netstat
- weitere Goodies ...

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„top“ - der Klassiker

```
top - 14:54:19 up 6 days, 21:35,  2 users,  load average: 3.84, 4.16, 3.89
Tasks: 163 total,   3 running, 160 sleeping,   0 stopped,   0 zombie
Cpu(s): 31.5%us,  6.2%sy,  0.0%ni, 48.5%id, 13.8%wa,  0.0%hi,  0.1%si,  0.0%st
Mem:   8190632k total,  7286972k used,   903660k free,   535488k buffers
Swap:  2104472k total,      108k used,  2104364k free,  3338780k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
18580 wwwrun    15   0  371m  90m 3984 R   61  1.1   0:09.99 httpd2-prefork
18618 wwwrun    16   0  375m  92m 3884 S   59  1.2   0:02.51 httpd2-prefork
 3683 mysql     15   0 1001m 742m 5896 S   51  9.3   3043:30 mysqld
18624 wwwrun    15   0  353m  70m 3648 R   32  0.9   0:00.97 httpd2-prefork
18610 wwwrun    15   0  384m 100m 5068 S   23  1.3   0:01.90 httpd2-prefork
18548 wwwrun    16   0  357m  74m 5208 S   21  0.9   0:05.98 httpd2-prefork
18623 wwwrun    15   0  352m  70m 3580 S   20  0.9   0:01.35 httpd2-prefork
18566 wwwrun    15   0  356m  73m 5236 S    9  0.9   0:11.85 httpd2-prefork
18620 wwwrun    15   0  300m  19m 3692 S    6  0.2   0:00.17 httpd2-prefork
18621 wwwrun    15   0  299m  17m 3264 S    5  0.2   0:00.14 httpd2-prefork
18579 wwwrun    15   0  349m  67m 3816 S    4  0.8   0:02.52 httpd2-prefork
```

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Load

```
top - 14:54:19 up 6 days, 21:35,  2 users,  load average: 3.84, 4.16, 3.89
Tasks: 163 total,   3 running, 160 sleeping,   0 stopped,   0 zombie
Cpu(s): 31.5%us,  6.2%sy,  0.0%ni, 48.5%id, 13.8%wa,  0.0%hi,  0.1%si,  0.0%st
```

- load average: Exponentiell geglätteter Wert der durchschnittlichen Anzahl von Prozessen in der „run queue“ über einen Zeitraum von 1, 5 und 15 Minuten

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Load

- Was ist eine hohe Load?
  - Jedes System reagiert anders
  - Abhängig von Aufgabe und Anzahl der CPU-Kerne im System, kann eine Load von 0,5 schon als „träge“ empfunden werden und eine Load von 12 „normal“ sein
- Was bringt uns dann die Load?
  - Kapazitätsplanung

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

CPU-Zeit

```
top - 14:54:19 up 6 days, 21:35,  2 users,  load average: 3.84, 4.16, 3.89
Tasks: 163 total,   3 running, 160 sleeping,   0 stopped,   0 zombie
Cpu(s): 31.5%us,  6.2%sy,  0.0%ni, 48.5%id, 13.8%wa,  0.0%hi,  0.1%si,  0.0%st
```

- us: user mode – Rechenzeit der Applikationen
- sy: kernel mode – Rechenzeit des Kernels
- ni: nice – nice Prozesse im user mode
- id: idle – Däumchen drehen
- wa: wait I/O – Warten auf die Fertigstellung von I/O
- hi, si: hardware- & software interrupts
- st: steal time – Gestohlene Zeit (XEN Hypervisor)

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Top Spalten

```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
18580 wwwrun    15   0  371m  90m 3984 R   61  1.1   0:09.99 httpd2-prefork
18618 wwwrun    16   0  375m  92m 3884 S   59  1.2   0:02.51 httpd2-prefork
 3683 mysql     15   0 1001m 742m 5896 S   51  9.3   3043:30 mysqld
```

- PR: priority – Je höher, desto länger hat der Prozess gewartet
- NI: nice value – Je niedriger, desto höher die Priorität (ungleich PR)
- VIRT: virtual image – Gesamtgröße des benutzten Speichers (inkl. shared libraries, Datensegmente, swap, ...)
- RES: resident size – „echter“ RAM-Verbrauch
- SHR: shared mem – gemeinsam genutzter Speicher (beispielsweise shared libraries)

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„free“ - Speicherauslastung

```
             total       used       free     shared    buffers     cached
Mem:       8190632    6725324    1465308          0     500632    2919244
-/+ buffers/cache:    3305448    4885184
Swap:      2104472        108    2104364
```

- Ungenutzter Speicher ist verschwendeter Speicher – nicht benötigter RAM wird für den Cache verwendet
- buffers/cached: gibt an, wieviele Bytes zum Cachen von Festplattenblöcken bzw. Dateien und Verzeichnissen verwendet wird
- buffers/cached wird bei Bedarf vom Kernel für Applikationen freigegeben
- ABER: ein System mit wenig Cache bremst das System

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Speicherlimitierung auf 32-bit CPUs

- 32-bit CPUs können nur 4 GB RAM adressieren

  $2^{32} = 4294967296$

- CPUs mit PAE (ab Pentium Pro) haben 36 Bit für die Adressierung von RAM => die CPU kann 64 GB adressieren
- 32-bit Applikationen können aber NIE mehr als 4 GB adressieren
  - In der Praxis i.d.R. nur 3 GB – dazu später mehr.

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Mehr als 1024 MB RAM bei Linux (32-bit)

- Low Memory und High Memory
  - ZONE_DMA: 0 bis 16 MB
  - ZONE_NORMAL: 16 bis 896 MB
  - ZONE_HIGHMEN: 896 bis ...

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Low Memory und High Memory

- `cat /proc/meminfo | egrep '(Low|High)'`

```
HighTotal:     3080168 kB
HighFree:        39680 kB
LowTotal:       880404 kB
LowFree:        406568 kB
```

- Alle Kernelstrukturen müssen im Low Memory Bereich liegen
- Der Kernel kann nur den Low Memory Bereich direkt ansprechen
- High Memory muß vor Benutzung im Low Memory Bereich „eingeblendet“ werden => Overhead und Performanceverlust

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Memory Split ab Kernel 2.6.13

- Memory Split 3/1, 2/2 oder 1/3 (user/kernel)
- `cat /proc/meminfo | egrep '(Low|High)'`

```
HighTotal:     1039808 kB
HighFree:       563208 kB
LowTotal:      2066672 kB
LowFree:       2031804 kB
```

- Applikationen können dann maximal 3, 2 bzw. 1 GB groß werden

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Low Memory und High Memory

- Bei Systemem mit viel RAM kann der Low Memory Bereich voll sein, ohne dass der High Memory Bereich voll ausgeschöpft werden kann
- 64-bit Systeme haben diese Trennung nicht. Es gibt nur Low Memory

  => Bei mehr als 1 GB RAM; spätestens aber ab 4 GB RAM immer 64-bit CPUs verwenden

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„pmap“ - Speichersünder aufspüren

- Speicheradresse | Größe | Zugriffsrechte | Ressource

```
6fac7000      8K r-x--  /usr/lib/apache2/modules/mod_speling.so
6fac9000      4K r----  /usr/lib/apache2/modules/mod_speling.so
6faca000      4K rw---  /usr/lib/apache2/modules/mod_speling.so
6facb000      8K r-x--  /usr/lib/apache2/modules/mod_setenvif.so
6facd000      4K r----  /usr/lib/apache2/modules/mod_setenvif.so
6face000      4K rw---  /usr/lib/apache2/modules/mod_setenvif.so
6facf000     56K r-x--  /usr/lib/apache2/modules/mod_rewrite.so
6fadd000      4K r----  /usr/lib/apache2/modules/mod_rewrite.so
6fade000      4K rw---  /usr/lib/apache2/modules/mod_rewrite.so
```

- Die Datensegmente (rw---) gehören dem Prozess. Der Rest kann mit anderen Prozessen geteilt werden
- Jedes Modul braucht privaten Speicher
  - Unbedingt alle Dienste auf ein Minimum abspecken
  - Selbstkompilierte Dienste können bei großen Systemen Ressourcen sparen

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„slabtop“ – Anzeige der Caches

OBJS	ACTIVE	USE	OBJ SIZE	SLABS	OBJ/SLAB	CACHE SIZE	NAME
345600	345519	99%	0.77K	69120	5	276480K	ext3_inode_cache
315913	315664	99%	0.20K	16627	19	66508K	dentry_cache
261720	123585	47%	0.09K	6543	40	26172K	buffer_head
54628	39079	71%	0.52K	7804	7	31216K	radix_tree_node
20001	16342	81%	0.06K	339	59	1356K	size-64
14432	12158	84%	0.17K	656	22	2624K	vm_area_struct
2904	2848	98%	0.09K	66	44	264K	sysfs_dir_cache
2895	2050	70%	0.25K	193	15	772K	filp
2880	843	29%	0.02K	20	144	80K	anon_vma
2118	2031	95%	0.60K	353	6	1412K	proc inode cache

- Der Cache ist in sogenannte „slabs“ organisiert
- slabtop zeigt die Größe und Auslastung der einzelnen slabs an

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„lsof“ - list open files

- `lsof -p PID`

  zeigt alle benutzten Ressourcen des Prozesses
- `lsof /datei`

  zeigt alle Programme an, die die Datei benutzen
- `lsof -i`

  zeigt alle bnenutzen IP-Sockets an

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Beispielausgabe von lsof

```
COMMAND  PID USER   FD   TYPE DEVICE    SIZE   NODE NAME
bash    6831 root  cwd    DIR    0,3       0   8925 /proc/6831
bash    6831 root  rtd    DIR    3,3    4096      2 /
bash    6831 root  txt    REG    3,3  651148 204795 /bin/bash
bash    6831 root  mem    REG    3,3   34244 264537 /lib/libnss_file
bash    6831 root  mem    REG    3,3   34280 264378 /lib/libnss_nis-
bash    6831 root  mem    REG    3,3   79544 264567 /lib/libnsl-2.6.
bash    6831 root  mem    REG    3,3   26260 264509 /lib/libnss_comp
bash    6831 root  mem    REG    3,3 1237276 264510 /lib/libc-2.6.1.
bash    6831 root  mem    REG    3,3    9612 264499 /lib/libdl-2.6.1
bash    6831 root  mem    REG    3,3  266136 271961 /lib/libncurses.
bash    6831 root  mem    REG    3,3  108996 264500 /lib/ld-2.6.1.so
bash    6831 root    0u   CHR  136,0          2 /dev/pts/0
bash    6831 root    1u   CHR  136,0          2 /dev/pts/0
bash    6831 root    2u   CHR  136,0          2 /dev/pts/0
bash    6831 root  255u   CHR  136,0          2 /dev/pts/0
```

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

lsof

- FD (File Descriptor)
  - cwd – current working directory
  - mem – memory mapped file
  - rtd – root directory
  - txt – program text (das ausgeführte Programm)
  - rX, wX, uX – read, write bzw. read und write Zugriff auf den File Descriptor X

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

lsof

- TYPE
  - IPv4, IPv6 – IPv4 bzw. IPv6 Socket
  - unix – UNIX domain socket (beispielsweise /dev/log)
  - BLK – block special file (beispielsweise /dev/sda)
  - CHR – character special file (beispielsweise /dev/null)
  - DIR – directory
  - REG – regular File (jede „normale“ Datei)

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

lsof

- DEVICE
  - Major und Minor Number von Geräten (beispielsweise 3,1 für /dev/hda1)
  - Speicheradresse für Sockets
  - Referenznummer des Kernels für Ressourcen
- SIZE
  - Größe der Ressource in Bytes
- NODE
  - inode der Ressource

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„strace“ - trace system calls

- Nur der Linux Kernel kann direkt auf Ressourcen zugreifen und sie manipulieren (Speicherinhalte ändern, auf Dateien zugreifen, ...)
- Alle Programme im user space müssen dem Kernel daher durch sogenannte „system calls“ mitteilen was sie tun möchten
- strace kann uns sämtliche system calls anzeigen, die ein Programm aufruft

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Wichtige Systemaufrufe

- Datei- und Verzeichniszugriffe
  - open, close, read, write, chdir, mkdir, rmdir, rename, unlink, ...
- Datei- und Vezeichnisattribute auswerten und ändern
  - chmod, chown, stat, fstat, access, ...
- Netzwerkzugriffe
  - socket, bind, listen, accept, connect, send, ...
- Erzeugen und Beenden von Prozessen
  - execve, fork, vfork, exit, ...
- Speicherzugriffe
  - mmap2, brk, mprotect, ...

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

strace Syntax

- `strace PROGRAMM`

  startet und verfolgt Programm
- `strace -p PID`

  dockt sich an den Prozess an
- `strace -e SYSTEMCALL,SYSTEMCALL PROGRAMM`

  strace zeigt nur gelistete system calls an
- `strace -f PROGRAMM`

  Auch Unterprozesse werden verfolgt

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„iostat“ - input/output statistics

- iostat zeigt beim Aufruf die durchschnittliche Auslastung des Systems (ähnlich wie top) und die durchschnittliche I/O-Auslastung seit dem letzten Bootvorgang an

```
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          23.08    0.01    3.78    0.52    0.00   72.60

Device:            tps   Blk_read/s   Blk_wrtn/s   Blk_read   Blk_wrtn
sda               5.63        31.28       103.22    9425128   31106140
sdb              57.72       667.71      1759.25  201219584  530166592
dm-0              3.46         4.74        25.78    1427002    7769536
dm-1             10.04        26.52        77.44    7993010   23336320
dm-2            183.11       603.01      1422.75  181723170  428757248
dm-3             46.36        64.69       336.51   19495154  101409344
```

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

iostat

- Im Intervallmodus zeigt iostat die Differenzen zum letzten Intervall an
- `iostat -x 10`

  Zeigt erweiterte Angaben (-x) zur I/O-Auslatung an und generiert alle 10 Sekunden einen neuen Report

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

iostat

- tps – transfers per second
- r/s bzw. w/s – Lese- bzw. Schreibzugriffe pro Sekunde
- avgrq-sz – Durchschnittsgröße der Requests in Sektoren
- avgqu-sz – Durchschnittsgröße der Warteschlange des Geräts
- await – Durchschnittliche Wartezeit bis zur Erfüllung des I/O-Requests in ms
- rrqm/s bzw. wrqm/s – Anzahl der Lese- bzw. Schreibzugriffsaufträge pro Sekunde, die zusammengeführt wurden (das System gruppiert die Requests um performanter arbeiten zu können)

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„iotop“ - I/O Monitor

```
Total DISK READ: 692.63 K/s | Total DISK WRITE: 0 B/s
  PID USER      DISK READ  DISK WRITE   SWAPIN     IO>    COMMAND
 5673 root      692.63 K/s       0 B/s  0.00 % 98.52 % find /
    1 root            0 B/s       0 B/s  0.00 %  0.00 % init [2]
    2 root            0 B/s       0 B/s  0.00 %  0.00 % [kthreadd]
    3 root            0 B/s       0 B/s  0.00 %  0.00 % [migration/0]
    4 root            0 B/s       0 B/s  0.00 %  0.00 % [ksoftirqd/0]
```

- iotop funktioniert wie „top“ für I/O
- iotop erfordert spezielle Kerneloptionen
  - CONFIG_TASK_IO_ACCOUNTING
  - CONFIG_TASKSTATS
- Serienmäßig bieten das bisher nur Debian Lenny und Fedora 10

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

„netstat“ - network connections and interface statistics

- `netstat -u` bzw. `-t`

  UDP- bzw. TCP-Sockets anzeigen
- `netstat -l`

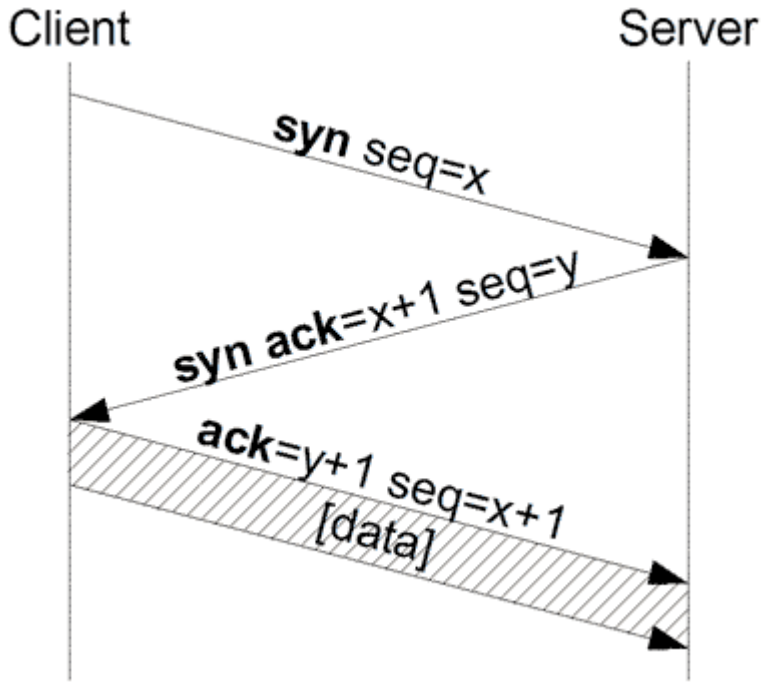
  nur Serversockets anzeigen
- `netstat -a`

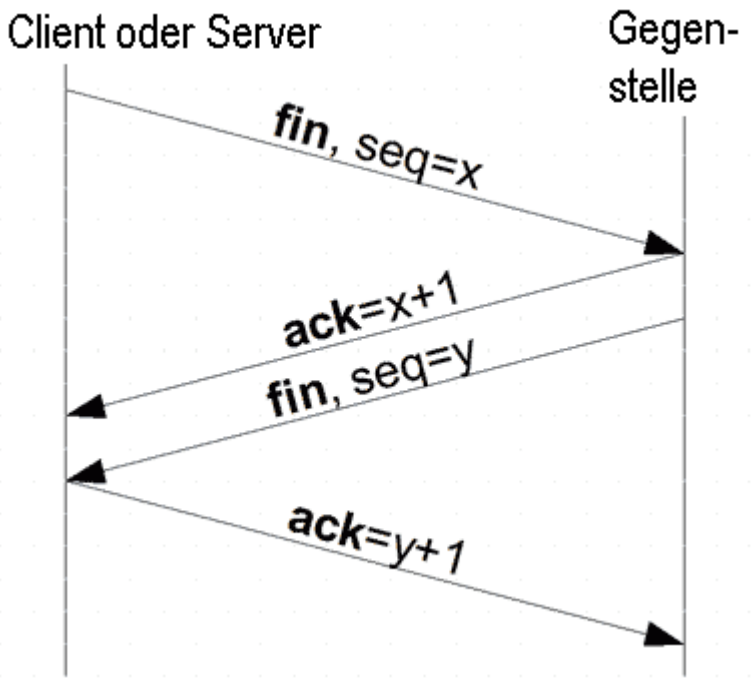
  zeigt Serversockets und Verbindungen an
- `netstat -p`

  zeigt zugehörige Prozesse an

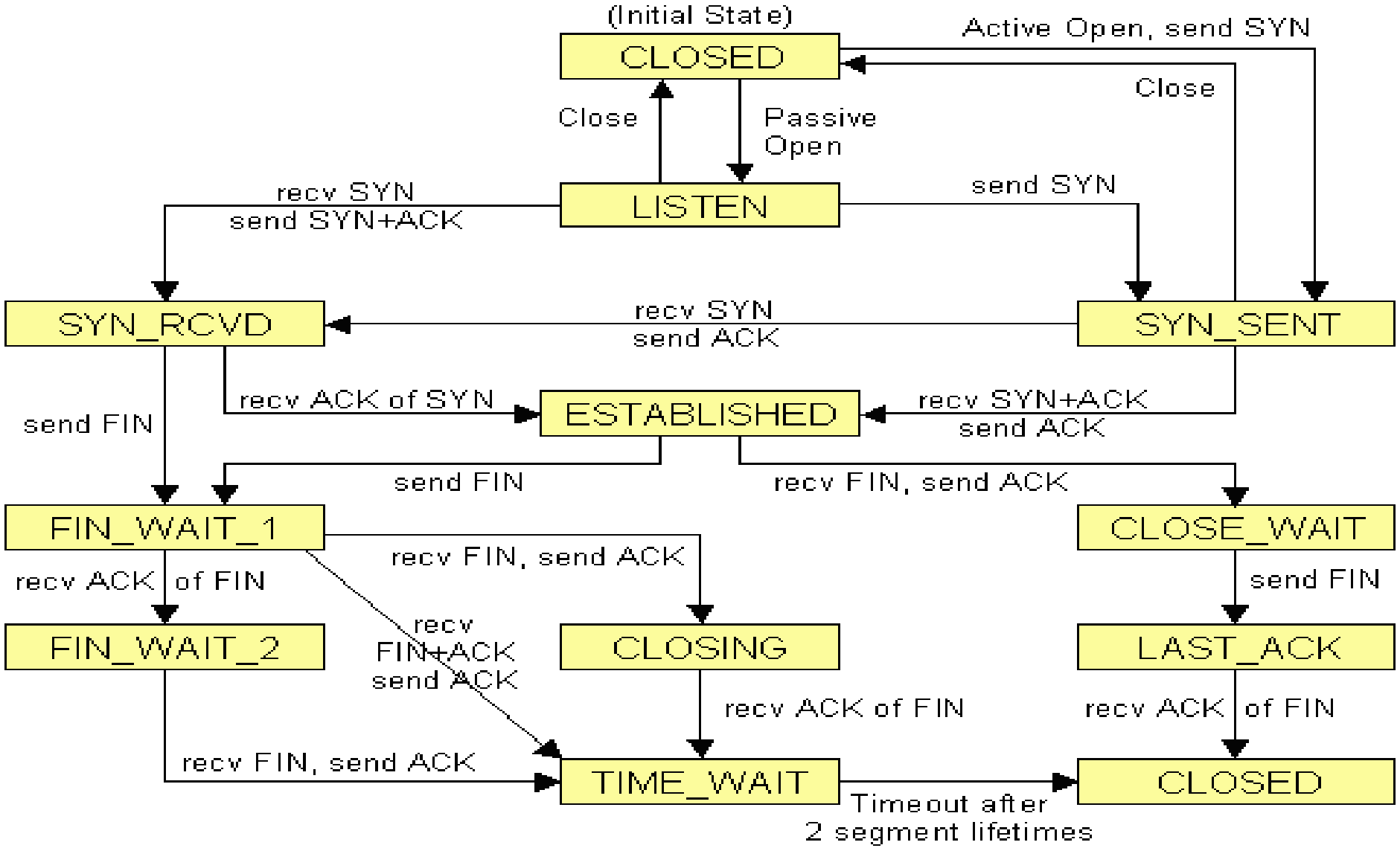
HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

TCP Handshake und Teardown





TCP State Diagram



HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Die missverstandenen States

- TIME_WAIT
  - Im TIME_WAIT-Zustand wird auf Pakete gewartet, die evtl. noch im Netzwerk sind. Dadurch wird verhindert, dass ein verspätet eintreffendes Paket eine neue Verbindung auf dem gleichen Socket stört.
  - TIME_WAIT dauert in der Regel 2 Minuten (2 x Maximum Segment Lifetime). Danach kann kein verspätetes Paket mehr eintreffen.
  - Die Applikation hat die Kontrolle bereits an das OS bzw. den IP-STACK abgegeben. TIME_WAIT blockiert demnach keine Ressourcen der Applikation.
  - TIME_WAIT ist gut ;)

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Die missverstandenen States

- CLOSE_WAIT
  - Das lokale Ende der Verbindung hat ein FIN erhalten und mit einem ACK bestätigt.
  - Die Applikation hat die Kontrolle über den Socket noch nicht abgegeben. Daher kann der IP-Stack noch kein FIN senden.
  - CLOSE_WAIT ist kein Problem des IP-Stacks. Der „Timeout“ kann nicht konfiguriert werden. Es ist ein Fehler in der Applikation!

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Weitere nützliche Tools

- iptraf – Tool zur Anzeige des Traffics auf einem Interface
- tcpdump/tshark & wireshark – Network Sniffer und Paket Analyzer
- mtr – traceroute Alternative
- ethtool – Setzen und Auslesen der Einstellungen für Netzwerkkarten
- powertop – Leistungsverbrauch auf Intel basierten Laptops

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Und nun ...

- Herzlichen Dank für Ihren Besuch und Ihre Aufmerksamkeit !

- Stefan Semmelroggen
- s.semmelroggen@heinlein-support.de
- Telefon: 030 / 40 50 51 - 0

HEINLEIN
LINUX: AKADEMIE · SUPPORT · HOSTING

Heinlein Professional Linux Support GmbH

- AKADEMIE
  - Von Profis für Profis: Wir vermitteln die oberen 10% Wissen. Geballtes Wissen und umfangreiche Praxiserfahrung aus erster Hand.
- SUPPORT
  - LPIC-2-Profis realisieren auch anspruchsvolle Projekte für Sie vor Ort. Im Heinlein CompetenceCall lösen sie Notfälle, auf Wunsch mit garantierten 24/7-Verfügbarkeiten.
- HOSTING
  - Wenn Hosting kein Massengeschäft sein darf: Individuelles Business-Hosting mit perfekter Maintenance durch unsere Linux-Profis. Sicherheit und Verfügbarkeit werden bei uns groß geschrieben.