

Nov 11, 11 9:26

**syslog-HOWTO.txt**

Page 1/8

HOWTO zum Daemon "syslog" bzw. "syslog-ng" (System Logging)  
 (C) 2008-2011 T.Birnthaler/H.Gottschalk <howtos(at)ostc.de>  
 OSTC Open Source Training and Consulting GmbH, <http://www.ostc.de>

\$Id: syslog-HOWTO.txt,v 1.16 2011-11-11 08:26:04 tsbirn Exp \$

Diese Dokument beschreibt die Eigenschaften und Möglichkeiten des System-Logging-Daemons "syslog" und "syslog-ng" von Linux-Systemen.

## Inhaltsverzeichnis

- 1) Einleitung
- 2) Eigenschaften von "syslog"
- 3) Probleme mit "syslog"
- 4) Konfiguration
  - 4.1) Kategorie / Ursprungsdienst (facility)
  - 4.2) Priorität / Dringlichkeit (priority)
  - 4.3) Aktion (action)
  - 4.4) Standard-Verzeichnis für Logdateien
  - 4.5) Beispiele für Einträge in "/etc/syslog.conf"
  - 4.6) Ablauf einer Umkonfiguration
- 5) Syslog-NG (Next Generation, Balabit)
  - 5.1) Objekte
  - 5.2) Globale Objekt-Optionen ("options")
  - 5.3) Objekt-Quelle ("source")
  - 5.4) Objekt-Ziel ("destination")
  - 5.5) Objekt-Bedingung ("filter")
  - 5.6) Objekt-Log ("log")
  - 5.7) Templates und Makros
  - 5.8) Beispielskript "log2mysql.sh"
- 6) rsyslog (Rainer Gerhards)
  - 6.1) Eigenschaften
  - 6.2) Konfiguration
- 7) Links

## 1) Einleitung

-----  
 In jedem Betriebssystem laufen im Hintergrund viele sogenannte "Daemonen" (Dienste/Services), die nicht an ein Terminal gebunden sind. Ihre Meldungen (Debugging, Info, Warnung, Fehler) müssen aber trotzdem irgendwo ausgegeben oder aufgezeichnet werden, um sie später auswerten zu können oder um darin Fehler zu suchen. Dafür gibt es unter Linux einen zentralen Dienst namens "syslog", an den die Daemonen ihre Meldungen schicken können.

Typische Anwendungen des syslog-Daemons sind:

- \* Protokollieren von Vorgängen
- \* Monitoring von Systemen
- \* Netzwerkweite Integration vieler Log-Quellen in zentrales Repository

Der Dämon "/usr/bin/syslogd" erlaubt das Aufzeichnen/Protokollieren der Meldungen vom Kernel oder anderen Programmen in Dateien ("Logging"). So können System-Fehler oder sonstige Ereignisse aufgezeichnet und später analysiert werden. Er nimmt die Meldungen anderer Dienste entgegen und verarbeitet sie. Was er damit tut, wird in einer Datei namens "/etc/syslog.conf" konfiguriert.

Jede Meldung, die "syslog" bekommt, wird vom Absender mit einer KATEGORIE (Facility/Quelle/Ursprungsdienst) und einer PRIORITÄT (Priority/Dringlichkeit) klassifiziert. Weiterhin enthält Sie den NAMEN des Sendeprogramms und einen frei wählbaren TEXT. Der NAME wird auch TAG (String) genannt. Mit ihm identifiziert sich der sendende Daemon oder die sendende Anwendung (wird zum Präfix vor dem MeldungsTEXT).

- \* Kategorie: kern, mail, auth(priv), cron, daemon, local0-9, lpr, user
- \* Priorität: panic/emerg, alert, crit, err(or), warn(ing), notice, info, debug

Auf Basis dieser Informationen (und evtl. anderer) steuert "syslog" die weitere Verarbeitung der Meldungen über BEDINGUNGEN. Trifft auf eine Meldung eine der Bedingung zu (es dürfen beliebig viele zutreffen), dann kann sie:

- \* In eine Datei geschrieben (angehängt) werden: /PFAD/ZUR/DATEI  
(Datei muss bereits existieren!)
- \* In eine Datei geschrieben werden ohne Puffern: -/PFAD/ZUR/DATEI  
(Datei muss bereits existieren!)
- \* Auf allen Terminals eines Users ausgegeben werden: USERNAME
- \* Auf allen Terminals ausgegeben werden: \*
- \* Einem Kommando (ausführbare Datei) übergeben werden: | KMDO
- \* Über das Netz an einen Rechner geschickt werden: @HOSTNAME  
(muss per Konfiguration Meldungen annehmen)

Logdateien liegen unter Linux fast ausschließlich unter "/var/log" bzw. Unterverzeichnissen darin und habe häufig die Endung "\*.log" oder einen Präfix "log.". Typische Namen von Standard-Logfiles sind:

```
* /var/log/allmessages           # ALLE Logmeldungen
* /var/log/apache2/access.log    # Webserver-Zugriffe
* /var/log/apache2/error.log     # Webserver-Fehler
* /var/log/auth.log              # An/Abmeldungen (erfolgreich/fehlgeschlagen)
* /var/log/cups/access_log       # Druckaufträge
* /var/log/kern.log              # Kernel-Meldungen
* /var/log/localmessages         # ALLE lokalen Logmeldungen
* /var/log/mail.(log|warn|err|info) # Mail-Logmeldungen (Fehler, Warnungen, ...)
* /var/log/messages              # ALLE Logmeldungen
* /var/log/samba/log.nmbd        # Samba Namensauflösung
* /var/log/samba/log.smbd        # Samba Sharezugriffe
* /var/log/syslog                # Systemmeldungen
* /var/log/user.log              # Benutzermeldungen
```

Folgende Meldungen werden NICHT von "syslog" aufgezeichnet:

```
* Kernel-Meldungen beim System-Boot landen in einem speziellen Kernel-Puffer
  (auflisten per "dmesg")
* Lokale Terminal-Logins
  (auflisten durch "last", "lastb", "lastlog" und "faillog")
* Von Benutzern abgesetzte Kommandos
  (auflisten durch "lastcomm" falls Benutzer-Accounting aktiv)
```

## 2) Eigenschaften von "syslog"

```
* Im Zshg. mit Sendmail entstanden
* Ausgangsbasis: BSD-syslog (sehr alt: 1983)
* Sehr einfach gestrickt: UDP-Nachrichten im Klartext
* Protokoll-Definition: RFC 3154 + RFC 3195
* Verwendet Port 514
* Option "-r" (remote) -> Annahme beliebiger Nachrichten auf UDP-Port 514
```

```
      Nachricht (kurzer Text)
Client -----> Server/Daemon
Client -----> Relay -----> Server/Daemon
```

## 3) Probleme mit "syslog"

```
* Paket-Verlust möglich (verbindungslose Übertragung per UDP garantiert Paket-Ankunft nicht)
* Paket-Überflutung möglich (DoS = Denial of Service)
* Gefälschte Pakete möglich (keine Authentifizierung, alles auf Port 514 akzeptiert)
* Nachrichten unverschlüsselt übertragen (liegen im Klartext vor)
* Unflexible Filter: nur Kategorie (Ursprungsdienst) + Priorität (Dringlichkeit)
* Kategorie + Priorität teilweise uneinheitlich verwendet
* Ursprüngliche Quelle geht bei Weiterleitung (Relay) evtl. verloren
```

Lösung: "syslog-NG" oder "rsyslog" (siehe weiter unten).

## 4) Konfiguration

Welche Meldungen wo aufgezeichnet werden sollen, ist in folgender Konfigurations-Datei einzutragen:

```
/etc/syslog.conf
```

Alle Meldungen werden zusätzlich standardmäßig auf Konsole "tty10" ausgegeben (mit "Strg-Alt-F10" erreichbar).

Jede Zeile in der Konfigurations-Datei enthält ein oder mehrere durch ";" getrennte "KATEGORIE.PRIORITÄT"-Paare (Ursprungsdienst + Dringlichkeit) der aufzuzeichnenden Meldungen und dann durch TABs (keine Leerzeichen!) getrennt die beim Auftreten einer derartigen Meldung auszuführende Aktion "ACTION" (z.B. den Namen einer Datei für die Aufzeichnung der Meldungen).

```
KATEGORIE.PRIORITÄT;... TAB... ACTION
```

Mehrere solche Paare mit der gleichen Aktion sind durch ";" zu trennen (kein Leerzeichen dazwischen!). Mehrere Kategorien und/oder Prioritäten mit der gleichen Aktion sind durch "," zu trennen. Ein "=" vor einer Priorität verlangt GENAU diese Priorität, ein "!" vor einer Priorität trifft auf alle NIEDRIGEREN Prioritäten zu; ein "!=" vor einer Priorität trifft auf alle Prioritäten AUSSER dieser zu. Kommentarzeilen werden wie üblich durch ein führendes "#" gekennzeichnet, Leerzeilen sind ebenfalls erlaubt.

Nov 11, 11 9:26

**syslog-HOWTO.txt**

Page 3/8

Einschränkungen bei Solaris (und alten syslog-Versionen):

- \* Nur TABS zur Trennung von KATEGORIE.PRIORITÄT und ACTION erlaubt
- \* "\*" nur bei KATEGORIE erlaubt, nicht bei PRIORITÄT  
(Ersatz: "DEBUG" statt "\*" verwenden)

## 4.1) Kategorie / Ursprungsdienst (facility)


Kann einer der folgenden Werte (oder "\*" für alle) sein:

Wert	Nr	Bedeutung
kern	0	Kernel
user	1	Meldung von Benutzern
mail	2	Mailsystem
daemon	3	Daemon allgemein
auth	4	Anmeldung/Authentifizierung (früher "security")
syslog	5	Meldung durch syslog selbst
lpr	6	Drucksystem
news	7	Newsgruppensystem
uucp	8	UUCP-Meldung (Unix to Unix Copy)
cron	9	cron-Daemon (clock)
authpriv	10	Anmeldung/kritische Sicherheitsmeldungen
ftp	11	FTP-Server
ntp	12	NTP-Subsystem
audit	13	?
console	14	?
cron2	15	cron-Daemon
local0..7	16-23	Frei verwendbare Meldungen (eigene Programme/Dienste)
mark		Markierung durch syslog selbst (Zeitstempel/timestamp)
*		Alle Kategorien

## 4.2) Priorität / Dringlichkeit (priority)

Kann einen der folgenden Werte (oder "\*" für alle) annehmen, die Dringlichkeit steigt dabei von oben nach unten. Jede Priorität umfasst zugleich die HÖHEREN Prioritäten mit (d.h. "err" umfasst auch "crit", "alert", "emerg"):

Wert	Nr	Bedeutung
debug	7	Debugmeldung
info	6	Information
notice	5	Mitteilung
warn(ing)	4	Warnung
err(or)	3	Fehler
crit	2	Kritischer Fehler
alert	1	Alarm
emerg	0	Notmeldung (Kernel!)
panic	0	Notmeldung (VERALTET, nun "emerg")
*		Alle Prioritäten (analog "debug")
none		Kategorie NICHT aufzeichnen
=info		Meldungen GENAU dieser Priorität
!err		Alle Meldungen KLEINERER Priorität
!=alert		Alle Meldungen AUSSER "alert"


  
 zunehmende Dringlichkeit!

## 4.3) Aktion (action)

Kann einen der folgenden Werte annehmen. Wird der Dateiname mit einem "-" eingeleitet, so wird die Meldung sofort ("synchron") in die Datei geschrieben, d.h. NICHT zunächst im Speicher gepuffert. Stürzt der Rechner kurz nach dem Empfang einer ungepufferten Meldung ab, kann es daher NICHT passieren, dass die Meldung NICHT in der Logdatei steht:

Aktion	Bedeutung
FILE	An Datei FILE anhängen (gepuffert = asynchron, Pfadname)
-FILE	An Datei FILE anhängen (ungepuffert = synchron, Pfadname)
CMD	An Kommando "CMD" auf Stdin übergeben (Pfadname)
@HOST	An Rechner HOST schicken
USER,...	An Benutzer USER,... schicken (auf deren Terminals)
*	An alle Benutzer schicken (auf allen Terminals ausgeben)

## 4.4) Standard-Verzeichnis für Logdateien

-----  
 Das Standard-Verzeichnis für die Logging-Dateien lautet:

```
/var/log
```

Die Log-Dateien darin müssen regelmäßig vom Systemverwalter kontrolliert und gekürzt werden, um Systemprobleme festzustellen und das Dateisystem nicht mit Log-Meldungen "vollzumüllen". Alternativ übernimmt auch der Daemon "logrotate" diese Aufgabe.

## 4.5) Beispiele für Einträge in "/etc/syslog.conf"

```
-----
mail.*                /var/log/mail.log      # Alle Meldungen aus Mailsystem
mail.debug            /var/log/mail.log      # (analog)
mail,news.*          /var/log/mailnews.log  # Alle Meldungen von mail+news
auth.warning         /var/log/auth.log      # Sicherheits-M. ab "warning"
*.warn;*.err        /dev/tty10             # Warnungen+Fehler an Terminal tty10
*.crit               | lpr -P loglp        # Kritische Meldungen drucken
*.emerg;*.panic     *                      # Emergency+Panic an alle Term.
*.                  -/var/log/allmessages  # Alle Meldungen aufzeichnen
*.debug             -/var/log/allmessages  # (analog)
*.=info             /var/log/info.log      # NUR Info-Meldungen
*.!err              /var/log/belowerr.log  # Alle KLEINER Error-Meldungen
*.!=alert           /var/log/noalert.log   # Alle AUSSER Alert-Meldungen
```

## 4.6) Ablauf einer Umkonfiguration

-----  
 Um Änderungen am Log-Verhalten durchzuführen, müssen folgende Schritte in der angegebenen Reihenfolge durchgeführt werden:

- 1) Datei "/etc/syslog.conf" editieren (z.B. auskommentierte Zeile einkom.).
- 2) Neue Log-Dateien ANLEGEN, falls sie noch nicht existieren (z.B. per "touch /var/log/allmessages", unbedingt notwendig!).
- 3a) syslog-Dämon mitteilen, dass er seine Konfigurationsdatei neu lesen soll. Dazu ist seine Prozessnummer zu ermitteln und dem Prozess das Signal SIGHUP (1 = Hangup) zu schicken.

```
ps ax | grep syslogd      # => PID
kill -1 PID               # oder
kill -HUP PID             # oder
kill -SIGHUP PID          # oder
killall -HUP syslogd
```

- 3b) Alternativ kann der syslog-Dämon auch beendet und neu gestartet werden, da er bei jedem Start seine Konfigurationsdatei liest, oder er kann aufgefordert werden, seine Konfigurationsdatei neu zu lesen:

```
/etc/init.d/syslog restart
rcsyslog reload          # rc... nur bei SuSE-Linux
```

- 3c) Alternativ kann das System auch heruntergefahren und wieder neu gebootet werden, da der syslog-Dämon bei jedem Systemstart hochfährt und seine Konfigurationsdatei liest:

```
shutdown -r now
reboot
init 6
```

- 4) Eine ständig aktualisierte Anzeige aller Systemmeldungen auf einer Konsole wird z.B. durch Einloggen als root auf dieser Konsole und Absetzen des folgenden Kommandos erreicht (-f=follow):

```
tail -f /var/log/allmessages
```

- 5) Testen durch Erzeugen von Meldungen auf der Kommandozeile (Shell-Skript): (Default-Kategorie+Priorität "user.info" und Default Tag "logger")

```
logger -p KATEGORIE.PRIORITÄT -t TAG "TEXT"
logger "Hallo hier bin ich"
logger -p kern.panic "Nur ein Scherz..."
logger -p kern.panic -t TEST "Nur ein Test..."
```

## 5) Syslog-NG (Next Generation, Balabit)

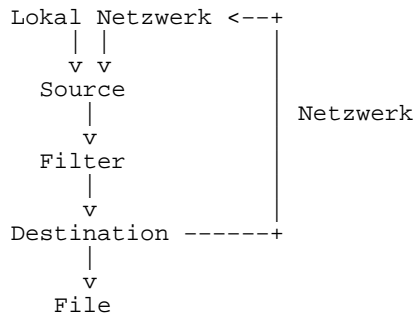
-----  
 \* Verbesserungen gegenüber "syslog"  
 + Netzwerkfähigkeiten wesentlich erweitert

- Auch per TCP (gesicherte Übertragung, verschlüsselte Tunnels)
- + Filter wesentlich erweitert
- + Logformat über "Templates" definierbar
- + Meldungsteile in Makros -> Automatismen
- + Feingranulare Logdateien
- + Innerhalb "file"-Destinations werden Makros expandiert
  - > automatische Host/Datums-Hierarchie von Verz./Dateien möglich

## \* Neue Version 3.0

- + OSE (Open Source Edition)
  - SSL/TLS-Verschlüsselung der Kommunikation
  - SSPMF-Support (Syslog Standard Protocol and Message Format)
  - Trennung in Name+Wert-Paare
  - Makros
  - Log-Meldungsteile umschreiben/modifizieren
  - Ablage in Datenbanken möglich (MySQL, MS-SQL, Oracle, PostgreSQL, SQLite)
- + PE (Premium Edition)
  - Kommerzielle Variante von Balabit (kostenpflichtig)
  - Sichere Ablage auf Platte (verschlüsselt, signiert)
  - Binäre und komprimierte Ablage auf Platte
  - Zeitstempel hochgenau und von externen Zeitquellen
  - Syslog-Agent für Windows
  - Windows-GUI für Nachrichtenkonfiguration
  - Nachrichten-Puffer auf Festplatte falls Logserver nicht erreichbar
  - Hochlastfähig
  - Viele Plattformen (Linux, Solaris, Windows, Open/Free/NetBSD, IBM AIX, HP-UX)

## \* Ablauf-Diagramm des syslog-ng:



## \* Konfigurationsdateien (entweder A oder B)

- /etc/syslog-ng/syslog-ng.conf # A) Für alle Dienste gemeinsam
- /etc/syslog-ng/syslog-ng.d/\* # B) Pro Dienst eine Datei

## 5.1) Objekte

Syslog-NG basiert auf einem "Baukasten" bestehend aus benannten "Objekten", die erst bei der Verknüpfung zu einem "Log-Objekt" aktiviert werden (auf Klammerung jedes Objekts durch "{...}" und Abschluss durch ";" achten!).

Objekttyp	Definitionssyntax
Option	options { OPTION1(PARAMS); ... };
Quelle/Source	source IDENT { DRIVER(PARAMS) };
Ziel/Destination	destination IDENT { DRIVER(PARAMS); ... };
Filter	filter IDENT { EXPRESSION; ... };
Log/Aktion	log { source(ID); ...; filter(ID); ...; destination(ID); ...; flags(FLAG; ...); };

## 5.2) Globale Objekt-Optionen ("options")

Legt globale Einstellungen des "syslog-ng"-Daemons fest.

Option + Parameter	Bedeutung
sync(N)	Anz. Pufferzeilen bevor auf Platte schreiben
log_fifo_size(N)	Anz. Ausgabepufferzeilen
chain_hostnames(Y/N)	Eigenen Hostnamen an generierenden anhängen
keep_hostnames(Y/N)	Umschreiben des Hostnamen aktivieren
check_hostnames(Y/N)	Hostnamen auf gültige Buchstaben prüfen
use_dns(Y/N)	DNS zur Hostnamenauflös. nutzen (blockiert!)
dns_ache(Y/N)	DNS-Antworten zwischenspeichern
dns_cache_size(N)	Anz. gepufferte DNS-Antworten
dns_cache_expire(N)	Anz. Sek die DNS-Antwort gepuffert wird

dns_cache_expire_failed(N)	Anz. Sek die fehlgeschlagene DNS-Antwort ...
use_fqdn(Y/N)	Vollständigen Hostnamen (FQHN) verwenden
stats(N)	Anz. Sek zwischen Statistiken

### 5.3) Objekt-Quelle ("source")

Definiert, welche Quelle von Meldungen relevant ist.

Quelle	Beispiel
unix-stream	unix-stream("/dev/log")
internal	internal() WICHTIG!
file	file("/proc/kmsg" log_prefix("kernel: "))
pipe/fifo	pipe("/dev/xconsole")
usertty	usertty("root")
udp	udp(ip(0.0.0.0) port(514))
tcp	tcp(ip(192.168.1.1) port(514) maxconnections(10))

### 5.4) Objekt-Ziel ("destination")

Definiert, was mit einer Meldung passieren soll (Speichern auf Datei, Übergeben an ein Skript/Programm, Weiterleiten an einen anderen Rechner).

Ziel	Beispiel
file	file("/var/log/syslog" owner("root") group("adm") perm(0755))
udp	udp("192.168.1.12") destport "514" spoof_source no)
tcp	tcp("192.168.1.12") destport "514" spoof_source no)
program	program("/usr/local/bin/log2mysql.sh")

### 5.5) Objekt-Bedingung ("filter")

Filtert über eine logische Kombination von Bedingungen aus den eintreffenden Meldungen die relevanten heraus. Filter sind verschachtelbar, d.h. ein Filter kann als Element andere Filter verwenden (per "filter(FILTERNAME)").

Folgende Filter-Funktionen gibt es:

Filter	Bedeutung
facility(FAC1, ...)	Quell-Kategorie(n)
level(PRI01, ...)	Quell-Priorität(en)
priority(PRI01, ...)	(analog)
program(REGEX)	Quell-Programm (per REGEX-Vergleich)
host(REGEX)	Quell-Rechner (per REGEX-Vergleich)
netmask(IP/MASK)	Quell-IP/Netzwerkmaske
match(REGEX)	Vergleich Meldungstext per REGEX
filter(NAME)	Subfilter-Name (Schachtelung von Filtern)

Bei "priority(...)" oder "level(...)" ist eine durch "," (Komma) getrennte Liste von Einzelprioritäten oder Bereichen der Form P1..PN (zwei Punkte) möglich.

Kombination der Filter-Funktionen erfolgt per:

Komb.	Bedeutung
(...)	Klammerung
and	Logischer UND-Operator
or	Logischer ODER-Operator
not	Logischer NICHT-Operator

Reguläre Ausdrücke zum Matchen der Nachrichtentexte in "match":

REGEX	Bedeutung
^	Meldungsanfang
\$	Meldungsende
.	EIN beliebiges Zeichen
[abc]	EIN Zeichen "a", "b" oder "c" (Zeichenklasse)
[a-c]	EIN Zeichen "a" ... "z" (Zeichenklasse)
[^abc]	EIN Zeichen AUSSER "a", "b" oder "c" (Zeichenklasse)
[^a-c]	EIN Zeichen AUSSER "a" ... "z" (Zeichenklasse)
R1 R2	Regexp R1 ODER Regexp R2
C*	0-N mal Zeichen C

Nov 11, 11 9:26

**syslog-HOWTO.txt**

Page 7/8

C?	0-1 mal Zeichen C
C+	1-N mal Zeichen C
\C	Metazeichen C ist normales Zeichen

Beispiele:

```
filter f_cnews { level(err,crit) and facility(news); };
filter f_iptables { facility(kern) and match("IN=") and match("OUT="); };
filter f_messages { not facility(news,mail) and not filter(f_iptables); };
```

## 5.6) Objekt-Log ("log")

Fasst Quelle + Filter + Ziel zusammen. Meldungen aus einer Quelle, die den Filter erfüllen werden dem Ziel übergeben.

- + Mehrere Quellen, Filter und Ziele erlaubt
- + Zusätzlich Optionen erlaubt (Flags)

Beispiele:

```
log { source(src); filter(f_syslog); destination(syslog); };
log { source(src); filter(f_daemon); destination(daemon); };
log { source(src); filter(f_lpr); destination(lpr); };
log { source(src); filter(f_mail); destination(mail); };
log { source(src); filter(f_user); destination(user); };
log { source(src); filter(f_all); destination(sql); };
```

## 5.7) Templates und Makros

Templates definieren Ausgabeformate von Meldungen, es sind darin Makros der Form \$XXX erlaubt. Die Option "template\_escape" sorgt dafür, dass Anführungszeichen maskiert werden.

Makro	Bedeutung
\$FACILITY	Quelle: auth(priv), cron, daemon, ftp, kern, lpr, mail, mark, news, security(=auth, nicht!), syslog, user, uucp, local0-7
\$PRIORITY	Priorität: debug, info, notice, warn(ing), err(or), crit, alert, emerg/panic
\$LEVEL	" "
\$TAG	Quelle+Priorität als 2-stellige Hexzahl
\$DATE	Datum im Standardformat
\$FULLDATE	Datum im Standardformat
\$ISODATE	Datum im Standardformat
\$YEAR	Jahr (4-stellig)
\$MONTH	Monat (2-stellig)
\$DAY	Tag (2-stellig)
\$WEEKDAY	Wochentag (3-stellig: Mon, Tue, ..., Sat, Sun)
\$HOUR	Stunde (2-stellig)
\$MIN	Minute (2-stellig)
\$SEC	Sekunde (2-stellig)
\$TZ	Zeitzone (3-stellig)
\$TZOFFSET	Zeitzone-Differenz zu GMT (5-stellig)
\$FULLHOST	Hostname (FQHN, mit Domain)
\$HOST	Hostname (ohne Domain)
\$PROGRAM	Programm von dem Meldung stammt
\$MESSAGE	Eigentlicher Nachrichteninhalt (Text inkl. Programmname + PID)
\$MSG	" "
\$MSGONLY	Nur Nachrichtentext

Beispiele:

```
template("[%YEAR/$MONTH/$DAY $HOUR:$MIN:$SEC] $PRIORITY $FACILITY $MESSAGE\n")

destination sql {
    program("/usr/local/sbin/log2mysql.sh"
    template("$HOUR $MIN $HOST $MSG")
    template_escape(yes)
};
```

## 5.8) Beispielskript "log2mysql.sh"

Beispiel für ein Programm "log2mysql.sh" zur Verarbeitung von Log-Meldungen (Ablage der Meldungen in einer MySQL-Datenbanktabelle):

```
#!/bin/sh
#-----
# log2mysql.sh
#-----
# destination sql {
#   program("/usr/local/sbin/log2mysql.sh"
#   template("$HOURL $MIN $HOST $MSG")
#   template_escape(yes)
# };
#-----
while read HOUR MIN HOST MSG
do
    echo "INSERT INTO logbook(hour,minute,host,message)
        VALUES ('$HOURL', '$MIN', '$HOST', '$MSG') " |
    mysql --user=USER --password=GEHEIM logs
done
```

## 6) rsyslog (Rainer Gerhards)

### 6.1) Eigenschaften

- + Templates möglich (Ausgabeformat der Meldung anpassen)
  - Drei verschiedene Ausgabeformate fest verdrahtet
  - Default: traditionelles Ausgabeformat von "syslog"
    - TraditionalFormatWithPRI
    - \$template TraditionalFormatWithPRI,"%PRItext%:%timegenerated% %HOSTNAME% %%syslogtag%%msg:::
 :drop-last-lf%\n"
 \$template MyFormat,"%programname% meldet am %timegenerated%: %msg:::drop-last-lf% mit Prior
 ität %%syskigseveruty%\n"
 \*.\* -/var/log/syslog;TraditionalFormatWithPRI
- + Flexibel konfigurierbar
- + Filter
- + TCP-fähig (verlässlicher Weiterversand)
- + GSSAPI/TLS-fähig (SSL-Verschlüsselung beim Weiterversand)
- + Datenbank-Anbindung möglich (MySQL, PostgreSQL)
- + PHPLogCon (in DB gespeicherte Logdaten mit Web-Browser sichten)
- + Erweiterung von "syslog" (alles bekannte bleibt)
- + Multi-Threaded
- + Weg über Relays nachvollziehbar
- + Lokale Pufferung der Nachrichten falls Empfänger noch nicht bereit ist
- + ISO-8601-Zeitstempel mit Unterscheidung von Millisekunden und Zeitzoneinformationen

### 6.2) Konfiguration

```
/etc/rsyslog.conf
/etc/syslog.conf           # Include
/etc/syslogd.d/*.conf      # Include
/etc/syslogd.d/00_common.conf # Include
/etc/syslogd.d/01_mysql.conf # Include
/etc/syslogd.d/01_postgresql.conf # Include

$IncludeConfig
```

## 7) Links

* <a href="http://syslog-win32.sourceforge.net">http://syslog-win32.sourceforge.net</a>	Syslog-Implementierung für Windows
* <a href="http://www.balabit.com/network-security/syslog-ng/">http://www.balabit.com/network-security/syslog-ng/</a>	syslog-ng (Logging System)
* <a href="http://www.rsyslog.com/">http://www.rsyslog.com/</a> - Rainer Gerhards)	rsyslog (alternatives Logprogramm)
* <a href="http://www.phplogcon.org/">http://www.phplogcon.org/</a> ) (Webinterface zu Syslog)	phpLogCon (Webinterface zu Syslog)
* <a href="http://www.syslog.org/">http://www.syslog.org/</a>	Syslog-Wiki und Forum
* <a href="http://content.hccfl.edu/pollock/AUnix2/Logging.htm">http://content.hccfl.edu/pollock/AUnix2/Logging.htm</a>	Logging, Log File Rotation, and Syslog Tutorial
* <a href="http://www.loganalysis.org/sections/syslog/syslog-replacements/">http://www.loganalysis.org/sections/syslog/syslog-replacements/</a>	Syslog Replacements
* <a href="http://www.loganalysis.org/ing-News-and-Information/">http://www.loganalysis.org/ing-News-and-Information/</a>	Loganalysis (The System Log: Logging News and Information)
* <a href="http://www.wikidorf.de/reintechnisch/Inhalt/SyslogNGEinfuehrung">http://www.wikidorf.de/reintechnisch/Inhalt/SyslogNGEinfuehrung</a>	Einführung in syslog-ng
* <a href="http://home.datacomm.ch/prutishauser/texte/syslog-ng-de.txt">http://home.datacomm.ch/prutishauser/texte/syslog-ng-de.txt</a>	syslog-ng reference manual (deu)
* <a href="http://www.campin.net/syslog-ng/faq.html">http://www.campin.net/syslog-ng/faq.html</a>	Syslog-NG FAQ
* <a href="http://www.campin.net/syslog-ng/syslog.html">http://www.campin.net/syslog-ng/syslog.html</a>	What is Syslog-NG?