Rsyslog – deutsche Qualitätsarbeit für Linux

Roman Gächter
Trivadis AG
CH-8152 Glattbrugg

Schlüsselworte
Deutsche Qualitätsarbeit setzt sich durch. Aktuelle Linux Distributionen verwenden heutzutage alle Rsyslog für System Logging. Rsyslog ist der neue Standard für RHEL basierte Distributionen, darunter auch Oracle Linux ab den Versionen 6. Reiner Gerhards hat der OpenSource Gemeinde mit Rsyslog ein tolles Syslog Tool spendiert. Rsyslog zeichnet sich durch das modulare Design, die gute Performance und ausgezeichnete Sicherheits-Funktionalitäten aus.
Im Vortrag wird gezeigt wie eine zentrale Syslog Umgebung in einer Oracle Datenbank Cloud - RAC auf Oracle Linux - in der Automobilindustrie implementiert wurde. Dabei werden die Syslog Daten verschlüsselt übertragen.

Syslog Architektur mit TLS-Verschlüsselung

Die folgende Graphik zeigt beispielhaft die Übersicht einer Syslog-Infrastruktur. Für die Redundanz sind zwei Logserver vorhanden. Grundsätzlich senden alle Oracle-DB Systeme die Syslogs über den Rsyslog-Client verschlüsselt auf die beiden Log-Server. DB-Systeme mit älteren Linux Versionen senden Syslog noch über UDP unverschlüsselt.

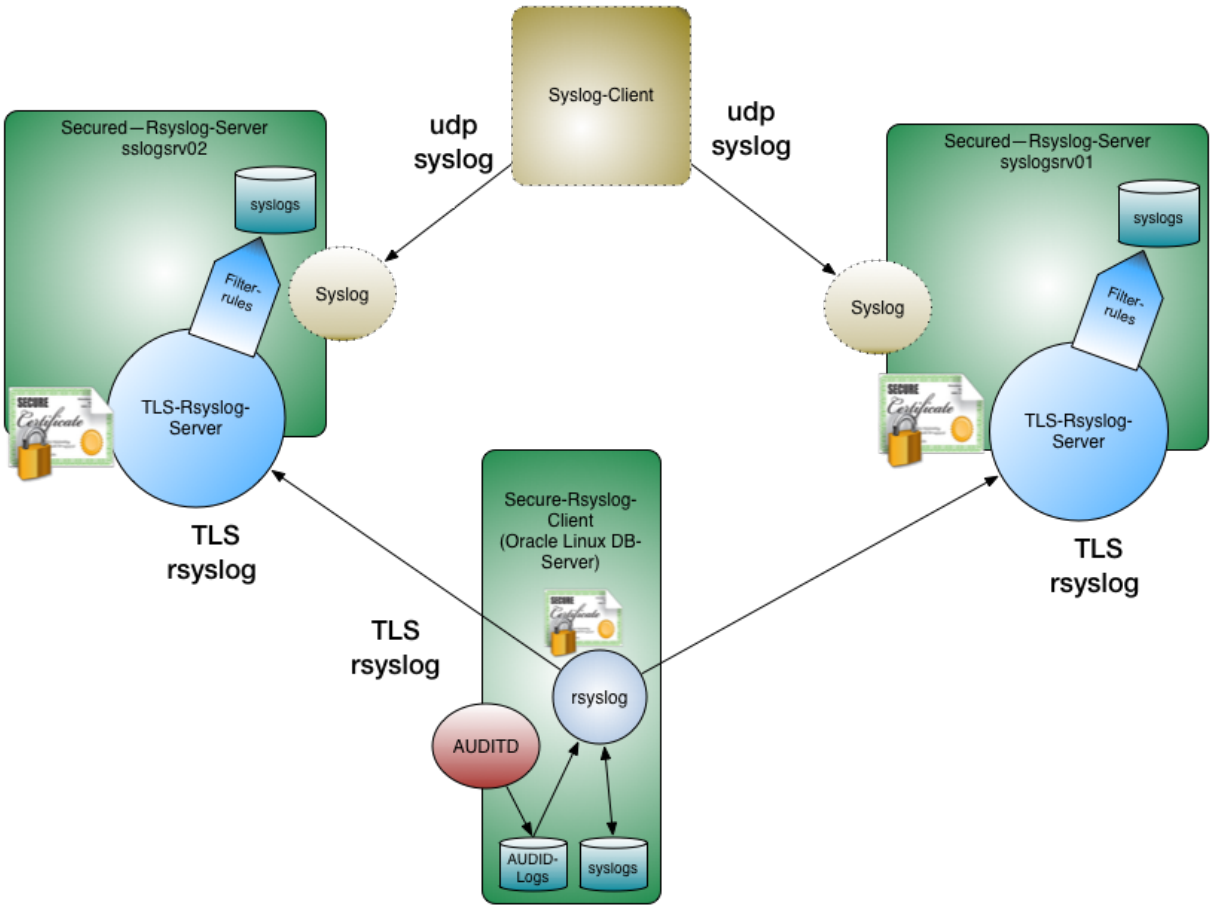

Abb. 1: Beispiel einer Architektur mit TLS-Verschlüsselung

Grundlagen zu Rsyslog / Syslog
Es existieren mehrere RFC's (Requests for Comments) für Syslog. Die unten aufgelisteten sind alle von Rsyslog implementiert:

- RFC 5424 - The Syslog Protocol (obsoletes RFC 3164)
  - https://tools.ietf.org/html/rfc5424
- RFC 5425 - Transport Layer Security Mapping for Syslog
- RFC 5426 - Transmission of Syslog Messages over UDP

Bei der Konfiguration von Rsyslog müssen diese RFC's zum Teil konsultiert werden, beispielsweise wenn Filter Regeln definiert werden sollen nach „Prioryties oder Facylities“. In den RFC's findet man die entsprechenden Zuordnungen.

Beispiel Zuordnung der „kernel“ Facility:

```
# Log all local kernel messages to the console and /var/log/kern
if $hostname == 'srv12' \
and $syslogfacility == '0' \
then /var/log/kern
```

In den RFC's sind die Syslog Facilities definiert. Facilities sind für die Konfiguration von Filterregeln notwendig und der Vollständigkeit halber in der folgenden Tabelle aufgelistet.

Facility Keyword	Description
auth, authpriv	Security, authentication, or authorization messages.
cron	crond messages
daemon	Messages from system daemons other than crond and rsyslogd
kern	Kernel messages
lpr	Line printer subsystem
mail	Mail system
news	Network news subsystem
syslog	Messages generated internally by rsslogd
user	User-level messages
UUCP	UUCP subsystem
local0 - local7	Local use

In den RFC's sind die Syslog Priorities definiert. Priorities sind für die Konfiguration von Filterregeln notwendig und der Vollständigkeit halber in der folgenden Tabelle aufgelistet.

Priority Keyword	Description
debug	Debug-level messages
info	Informational messages
notice	Normal but significant condition
warning	Warning conditions
err	Error conditions
crit	Critical conditions
alert	Immediate action required
emerg	System is unstable

Für die Facilities und Priorities muss bei Filterdefinitionen im Rsyslog der numerische Wert angegeben werden. Die numerischen Werte sind im RFC 5424 definiert und der Vollständigkeit halber in den folgenden beiden Tabellen aufgelistet:

```
Facility and Severity values are not normative but often used. They are
described in the following tables for purely informational    purposes.
Facility values MUST be in the range of 0 to 23 inclusive.

Numerical     Facility
Code
```

```
0       kernel messages
1       user-level messages
2       mail system
3       system daemons
4       security/authorization messages
```

```
5       messages generated internally by syslogd
6       line printer subsystem
7       network news subsystem
8       UUCP subsystem
9       clock daemon
10      security/authorization messages
11      FTP daemon
12      NTP subsystem
13      log audit
14      log alert
15      clock daemon (note 2)
16      local use 0  (local0)
17      local use 1  (local1)
18      local use 2  (local2)
19      local use 3  (local3)
20      local use 4  (local4)
21      local use 5  (local5)
22      local use 6  (local6)
23      local use 7  (local7)
```

```
Numerical         Severity
Code
0                 Emergency: system is unusable
1                 Alert: action must be taken immediately
2                 Critical: critical conditions
3                 Error: error conditions
4                 Warning: warning conditions
5                 Notice: normal but significant condition
6                 Informational: informational messages
7                 Debug: debug-level messages
```

Rsyslog

Redhat basierte Linux Distributionen – darunter auch „Oracle Linux“ - verwenden ab RHEL 6 per default das Rsyslog Packet für das System-Logging. (Auch in den aktuellen Distributionen von Debian und SLES ist Rsyslog der default Syslog.) Rsyslog bietet über die standard Syslog Funktionalitäten einige nützliche Erweiterungen zum standard Logging über Syslog UDP an:

- Skalierbarkeit
- Modularer Aufbau
- dedizierte TCP-Ports für Performance Optimierung
- Security
- Logging über TCP
- Verschlüsselung mit TLS
- Intelligente Rulesets
- Filtermöglichkeiten
- Integration von lokalen nicht Syslog Logfiles
- Anbindungen an Datenbanken
- Support für RFC 5424, RFC 5425, RFC 5426
- Support für RELP

- support for buffered operation modes where messages are buffered locally if the receiver is not ready

SW Distributions
Für Oracle Linux 5.x existieren parallel die Rsyslog Versionen 3.22.1 und 5.5.8 im Repository. Bei Oracle Linux 6.x kann man zwischen der Version 5.8.10 oder 7.4.10 wählen, bei beiden wird RELP unterstützt. OL 7.1 wir mit Rsyslog der Version 7.4.7 ausgeliefert, RELP wird unterstützt.

TLS mit RELP
Ab den Rsyslog Versionen > 5.8 ist das RELP bei Oracle Linux implementiert. Das RELP ist ein Protokoll, welches Datenverlust beim Unterbruch von Rsyslog TCP Verbindungen verhindern kann. Dies wird durch Transaktionsnummern und Bestätigung des Empfängers bewerkstelligt. Eine Rsyslog Transaktion kann beispielweise beim Restart des Rsyslog-Servers nach Logrotation Tasks unterbrochen werden. Durch die RELP Transaktionsnummern kann eine solche abgebrochene Transaktion nachgeführt werden, wenn der Server wieder erreichbar ist.

Fehlermeldungen von Rsylog nach Verbindungsabbruch:
Im Gegensatz zum Betrieb von Rsyslog mit UDP (es werden keine Fehler angezeigt bei Verbindungsabbrüchen, Log-Daten gehen verloren) werden beim Betrieb über TCP bei Unterbrüchen Fehler gelogged. Dies kann passieren wenn nach Rsyslog Konfigurationsänderungen ein Restart erfolgt oder nach Logrotation der Logfiles.

```
Jan 13 11:08:38 srv105 rsyslogd-2078: netstream session 0x4988510 will be
closed due to error
```

Mit RELP kann dies vermieden werden.

Basisstruktur
Die „syslog messages“ werden von Rsyslog mit Hilfe von „input modules“ empfangen. Danach werden sie zu „ruleset“ weitergeleitet wo sie gemäss definierten Bedingungen verarbeitet werden. Wenn eine Regel passt wird die „message“ gemäss den definierten Aktionen verarbeitet, zum Beispiel in ein File oder eine DB geschrieben oder an einen „remote“ Host weitergeleitet.

Modules
Das Design von Rsyslog erlaubt es je nach notwendiger Funktionalität Module dynamisch zu laden. Im Konfigurationsfile muss die entsprechende Direktive angegeben werden.

```
$ModLoad MODULE_name
```

- Input modules gather messages from various sources. Input module names always start with the im prefix (examples include imfile and imrelp).
- Filter modules allow rsyslogd to filter messages according to specified rules. The name of a filter module always starts with the fm prefix.
- Library modules provide functionality for other loadable modules. rsyslogd loads library modules automatically when required. You cannot configure the loading of library modules.
- Output modules provide the facility to store messages in a database or on other servers in a network, or to encrypt them. Output module

```
   names always starts with the om prefix (examples include omsnmp and
   omrelp).
•  Message modification modules change the content of an rsyslog
   message.
•  Parser modules allow rsyslogd to parse the message content of
   messages that it receives. The name of a parser module always starts
   with the pm prefix.
•  String generator modules generate strings based on the content of
   messages in cooperation with rsyslog's template feature. The name of
   a string generator module always starts with the sm prefix.
```

Statement Types

Das alte Syslog Format ist immer noch nützlich für einfache Konfigurationen.

```
mail.info /var/log/mail.log
mail.err @server.example.net
```

Das „legacy rsyslog“ Format beginnt mit einem Dollar Zeichen.

```
$ModLoad imuxsock # provides support for local system logging (e.g. via
logger command)
$ModLoad imklog   # provides kernel logging support (previously done by
rklogd)
# Provides UDP syslog reception
$ModLoad imudp
$UDPServerRun 514
```

Das neue „style format“ für Rsyslog ist Rainer Script. Das beste Format für komplexe Konfigurationen.

```
if $msg contains 'Abort command issued' \
  or $msg contains 'blocked for more than 120 seconds' \
  or $msg contains 'checker reports path is down' \
  or $msg contains 'kernel: device-mapper: multipath: Failing path' \
  or $msg contains 'multipathd: checker failed path' \
  or $msg contains ': mark as failed' \
  or $msg contains 'checker reports path is up' \
  or $msg contains 'require eh work' \
  or $msg contains 'Error handler scsi_eh' \
  or $msg contains 'FCP command status' \
  or $msg contains 'aborting sp' \
  or $msg contains 'firmware reported underrun' \
  or $msg contains 'Dropped frame(s) detected' \
  or $msg contains 'kernel: end_request: I/O error' \
  or $msg contains 'error calling out /sbin/mpath_prio_alua' \
then /u00/rsyslog/applications/scripts/scsiaborts
```

Templates

Templates sind ein „key feature“ von Rsyslog. Sie erlauben es genau das Format zu spezifizieren, welches der Benutzer möchte. Templates werden auch dann verwendet, wenn dynamische Filenamen generiert werden müssen.

```
$template HostAudit,
"/u00/rsyslog/applications/auditd/%HOSTNAME%/audit_log"
$template auditFormat, "%msg%\n"
local5.*                                ?HostAudit;auditFormat
```

Message Properties

Die folgenden Message Properties sind nützlich für Filterregeln, bei der Definition von Templates und im Rainer Script.

```
msg - the MSG part of the message (aka “the message” ;))

rawmsg - the message excactly as it was received from the socket.
Should be useful for        debugging.

hostname - hostname from the message

source - alias for HOSTNAME

fromhost - hostname of the system the message was received from (in
a relay chain, this is the system immediately in front of us and not
necessarily the original sender). This is a DNS-resolved name,
except if that is not possible or DNS resolution has been disabled.

fromhost-ip - the same as fromhost, but alsways as an IP address.
Local inputs (like imklog) use 127.0.0.1 in this property.
```

System Properties
Die System Properties werden von der “rsyslog core engine” zur Verfügung gestellt. Sie sind unabhängig von der Message und beginnen mit einem $-Zeichen.

Beispiel von System Properties:

```
$myhostname - the name of the current host as it knows itself
(probably useful for filtering in a generic way)
$month
    The current month (2-digit)
$day
    The current day of the month (2-digit)
$hour
    The current hour in military (24 hour) time (2-digit)
```

Expression Based Filters
Im Folgenden ein Beispiel für einen „expression based filter“. Alles ausser den Facilities Mail, Cron und Authpriv mit Level info oder höher wird in's Logfile /var/log/messages geschrieben:

```
if $hostname == 'iumg105' \
  and not ( $syslogfacility == '2' ) \
  and not ( $syslogfacility == '4' ) \
  and not ( $syslogfacility-text == 'cron' ) \
```

```
then                /var/log/messages
```

TLS Verschlüsselung
Wenn man security relevante Daten, wie sie in Oracle Audit Logs und dem Output vom Linux Auditd Daemon vorkommen verarbeitet, ist es von Vorteil die Kommunikation zwischen Syslog-Client und -Server zu verschlüsseln. Dazu eignet sich die im Rsyslog eingebaute TLS-Verschlüsselung.
Voraussetzung dazu sind X509-Zertifikate, welche auf beiden Seiten installiert werden müssen.
Die Rsyslog TLS Verschlüsselung bietet folgende Funktionalitäten:

- Verschlüsselung des Datenverkehrs über das Netzwerk
- Der Syslog Sender authentifiziert den Syslog Empfänger, der Empfänger kontrolliert wer senden darf
- Der Syslog Empfänger authentifiziert den Syslog Sender, der Sender kann überprüfen ob er wirklich mit dem erwarteten Empfänger kommuniziert
- Die gegenseitige Authentifizierung verhindert eine „man in the middel“ Attacke.

Zertifikate
Jeder Rsyslog Client benötigt ein signiertes x509 SSL Zertifikat. Auf dem Client müssen der Zertifikats-Schlüssel, das Zertifikat der CA „certificate authority“ und das von der CA signierte Zertifikat installiert werden.

Linux Audit Daten
In den Linux Auditd-Logs befinden sich security relevante Daten, welche nicht im Klartext – dies ist der Standard im Syslog UDP - übertragen werden sollten. Beispielsweise findet man in den Linux Auditd-Logs Informationen zu Benutzer Authentisierungen, Login Fehlern von SSH-Verbindungen mit direkten Hinweisen zu IP-Adressen und weitere heikle Informationen.

Einbinden von Auditd Logs
Das kann auf zwei Arten gemacht werden. Auditd direkt an Syslog anbinden oder das Auditd Logfile mit dem Rsyslog Module Imfile einlesen. Man könnte auch beides implementieren. Die Audit Logs wären dann einerseits in den normalen Syslogs enthalten andrerseits aber auch dediziert als Kopie des lokalen Audit-Logs. Der Vorteil dabei ist, dass die Auditd Logfiles unverändert auf den zentralen Syslog Servern übertragen werden. Somit kann man die standard Auswertungstool vom Auditd „ausearch“ und „aureport“ zum Auswerten der Logfiles verwenden.

Administration
Ein wichtiger Punkt in der Administration ist das Management der Logfiles. Es ist ja normalerweise nur ein begrenzter Speicherplatz vorhanden. Es sollte also ein Monitoring der Datenmenge vorhanden sein. Man sollte sich ein Houskeeping einrichten, dass so viele Daten wie möglich vorgehalten werden. Für das Housekeeping könnte man Logrotate verwenden.
Nach Änderungen in den Rsyslog Logfiles (Logrotate) oder nach Anpassungen der Rsyslog Konfiguration sind folgende Maßnahmen notwendig:
Änderungen in Logfiles:

```
kill -HUP `cat /var/run/rsyslogd.pid 2> /dev/null`
```

Änderungen in der Rsyslog Konfiguration:

```
service rsyslog restart
```

(Ein „service rsyslog reload“ funktioniert nicht, da wird keine Aktion ausgeführt).

Troubleshooting
Rsyslog im Debugging Mode starten:
(rsyslogd -dn, Restart des Rsslogd nach Debug Umstellung)

```
export RSYSLOG_DEBUGLOG="/path/to/debuglog"
export RSYSLOG_DEBUG="Debug"
rsyslogd –dn
```

Check der Rsyslog Syntax:

```
rsyslogd –N 1
```

Fazit
Rsyslog hat wohl wegen des modernen, modularen Designs und der guten Performance Einzug in die aktuellen Linux Distributionen gefunden. Mit Rainer Script kommt eine mächtige Scriptsprache mit, welche es dem Administrator ermöglicht sehr einfach komplexe Filterregeln einzubinden. Mit der Möglichkeit, die Kommunikation zum Logserver mit TLS zu verschlüsseln können auch sicherheitskritische Komponenten wie Audit-Logs in eine Syslog Infrastruktur integriert werden.

Kontaktadresse:
Roman Gächter
Trivadis AG
Europastrasse 5 Tel: +41-44-8087020
CH-8152 Glattbrugg Fax: +41-44-8087021