Exklusiv Online:
Wenn Programme nach Hause telefonieren
GFS2 und OCFS2, zwei Cluster-Dateisysteme im Linux-Kernel
Die Suche nach dem Groupware-Standard

Themengebiete
Desktop
Server
Netzwerk
Security
Administration
Entwicklung
Hardware
Szene
Special

Abo & Archiv
Linux.local
Stellenmarkt

Events
« August 2007 »

Mo	Di	Mi	Do	Fr	Sa	So
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Service
RSS-Feed abonnieren
Mediadaten
Kontakt

GFS2 und OCFS2, zwei Cluster-Dateisysteme im Linux-Kernel

Server-Traube

von Udo Seidel

Cluster-Dateisysteme waren vor Jahren proprietär und kostspielig. Mittlerweile bietet schon der Standard-Linux-Kernel mit dem Global Filesystem 2 (GFS2) und Oracles Clusterfilesystem 2 (OCFS2) zwei solche Dateisysteme zu GPL-Konditionen. Dieser Artikel gibt eine Einführung in ihre Funktionsweise und Einrichtung.

Wie können möglichst viele Rechner auf den gleichen - vielleicht sogar denselben - Datenbestand zugreifen? Eine Möglichkeit ist die Replikation der Daten über NFS, Rsync oder SnapMirror. Mittlerweile rückt die Nutzung von Storage Area Networks (SANs) immer mehr in den Mittelpunkt. Obwohl eigentlich im Netzwerk angesiedelt, erscheint der Speicher dabei wie ein lokaler Datenspeicher (Direct Attached Storage - DAS). Die Anbindung von SAN geschieht üblicherweise über Fibre Channel oder iSCSI, auch Infiniband ist möglich.

Für einen problemlosen Zugriff von mehreren Rechnern auf den Datenbestand kommen sogenannte Cluster-Dateisysteme zur Anwendung. Bekannte Vertreter aus dem propretären Unix-Lager sind das Clustered XFS (CXFS) von SGI oder das General Parallel Filesystem (GPFS) von IBM. Beide sind ebenfalls für Linux erhältlich. Seit einiger Zeit befinden sich Auslieferungsumfang des Linux-Kernels zwei GPL-lizenzierte Cluster-Dateisysteme: Das Global Filesystem 2 (GFS2) von Red Hat und Oracles Clusterfilesystem 2 (OCFS2).

Geschichtliches zu GFS2 und OCFS2

Das Global Filesystem entstand Mitte der 90er Jahre und ist damit ungefähr 10 Jahre älter als Oracles Cluster-Dateisystem. Die ersten Schritte machte GFS an der Universität von Minnesota. Die weitere Entwicklung erfolgte in der eigens dafür gegründeten Firma Sistina. Die Sistina-Entwickler trieben auch die Entwicklung des Logical Volume Managers (LVM) für Linux stark voran. Zunächst stand GFS unter der GNU Public License (GPL), 2001 änderte Sistina jedoch die Lizenz. Seit der Übernahme der Firma durch Red Hat steht GFS wieder unter der GPL. Hinter der Bezeichnung GFS2 steckt die durch Red Hat vorangetriebenen Weiterentwicklung von GFS. Die Aufnahme in den offiziellen Linux-Kernel erfolgte mit Version 2.6.19.

Oracle entwickelte zunächst das Cluster-Dateisystem OCFS mit dem Ziel, Datenbank-Dateien und Datenbank-Konfiguration in einem Cluster einfach verwalten zu können. Der Nachfolger OCFS2 legt diese Einschränkung ab und versteht sich als vollwertiges POSIX-kompatibles Dateisystem. Obwohl erst 2005 vorgestellt, ist OCFS2 schon seit Version 2.6.16 Bestandteil des offiziellen Kernels.

Cluster-Konfiguration

Die Software-Stacks von GFS2 und OCFS2 haben prinzipiell den gleichen Aufbau - die Kern-Komponenten sind Cluster-Software, Locking-Mechanismus und Fencing-Technik. Die Cluster-Software gibt die äußere Struktur vor. In der Cluster-Konfiguration legt der Admin fest, welche Knoten zum Cluster gehören und auf das Cluster-Dateisystem zugreifen können (siehe Listing 1). Für bestimmte Funktionen, beispielsweise das Locking, greifen sowohl GFS2 als auch OCFS2 auf die entsprechenden Cluster-Prozesse zurück.

Das grafische Tool ocfs2console hilft bei der Einrichtung von OCFS2.

Zum Anlegen der Cluster-Konfiguration stehen dem Anwender zwei Möglichkeiten zur Auswahl: der Lieblings-Editor oder grafische Werkzeuge. Bei GFS2 findet gerade die Ablösung von "system-config-cluster" durch das Tool Conga statt. Für OCFS2 verwendet der Anwender das Programm "ocfs2console" (Abbildung 1). Wichtig ist, dass die Konfiguration auf allen Cluster-Knoten identisch ist.

Listing 1: "/etc/ocfs2/cluster.conf"

```
node:
        ip_port = 7777
        ip_address = 192.168.0.1
        number = 0
        name = node0
        cluster = ocfs2
node:
        ip_port = 7777
        ip_address = 192.168.0.2
        number = 1
        name = node1
        cluster = ocfs2
cluster:
        node_count = 2
        name = ocfs2
```

Locking sorgt für klare Verhältnisse

Dass mehrere Cluster-Knoten auf dieselben Datenblöcke zugreifen, ist bei Cluster-Dateisystemen ein völlig normaler Vorgang. Zur Vermeidung von Inkonsistenzen im Dateisystem verwenden GFS2 und OCFS2 einen Distributed Lock Manager (DLM). Beide Implementierungen stammen vom VMS-DLM ab, bei OCFS2 ist allerdings eine stark vereinfachte Version enthalten.

Tabelle 1 zeigt die Kompatiblität konkurrierender DLM-Locks. Von den angegebenen Lock-Modi unterstützt OCFS2 nur den Exclusive Lock (EX), den Protected Read Lock (PR) und den No Lock (NL). GFS2 unterstützt zusätzlich den Concurrent Write Lock (CW), Concurrent Read Lock (CR) und Protected Write Lock (PW).

Bei bestehendem CW-Lock sind gleichzeitig weitere Locks der Modi NL, CR und CW möglich. Während sich der NL-Lock mit jedem anderen Lock verträgt, duldet der EX-Lock keinen weiteren Lock neben sich.

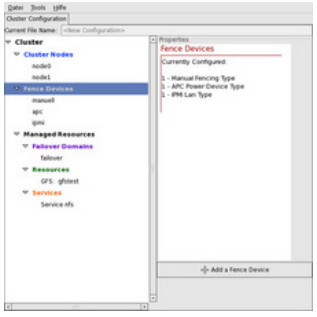
Tabelle 1: Kompatibilitäts-Matrix konkurrierender DLM-Locks.

	bestehender Lock					
angefragter Lock	NL	CR	CW	PR	PW	EX
NL	ja	ja	ja	ja	ja	ja
CR	ja	ja	ja	ja	ja	nein
CW	ja	ja	ja	nein	nein	nein
PR	ja	ja	nein	ja	nein	nein
PW	ja	ja	nein	nein	nein	nein
EX	ja	nein	nein	nein	nein	nein

Fencing schützt das Dateisystem

Über regelmäßge Heartbeats prüfen die beiden Cluster-Dateisysteme, ob alle Cluster-Knoten intakt sind. Ist ein Rechner nicht mehr erreichbar, muss das Cluster-Dateisystem verschiedene Aktionen durchführen, um die Konsistenz des Dateisystems sicherzustellen: Zunächst muss es verhindern, dass der "tote" Knoten weiterhin auf das Cluster-Dateisystem zugreift. Der einfachste Weg ist die Trennung des Cluster-Rechners vom Storage - das sogenannte Fencing. Es gibt zwei unterschiedliche Fencing-Ansätze. Beim I/O-Fencing, auch als Zoning bezeichnet, erfolgt die Trennung am Fibre-Channel-Switch durch Deaktiverung des entsprechenden Ports. Der Cluster-Knoten kann nicht mehr auf das SAN-Gerät zugreifen, bleibt aber selbst unangetastet. Dies erleichtert dem Admin die Fehleranalyse, warum der Rechner nicht mehr für Heartbeats erreichbar war. Die entsprechende Hardware vorausgesetzt, unterstützt nur GFS2 diese Fencing-Methode.
Die zweite Möglichkeit ist das Power-Fencing. Dabei erfolgt eine Abschaltung oder ein Reboot des "toten" Cluster-Knotens. OCFS2 unterstützt hier nur einen selbst ausgelösten Reboot oder, falls konfiguriert, eine Kernel-Panic, die den defekten Knoten zum Schutz des Dateisystems deaktiviert.

Die Wahlmöglichkeiten in GFS2 sind feiner: Neben dem manuellen Fencing ist es möglich, das Ausschalten/Rebooten über integrierte Management-Schnittstellen wie IPMI (Intelligent Platform Management Interface) oder iLO (integrated Lights Out) zu konfigurieren (Abbildung 2). Außerdem kann GFS2 einige Power-Switches ansteuern und den entsprechenden Rechner stromlos schalten. Die verschiedenen Fencing-Techniken kann der Admin auch kaskadierend konfigurieren und einsetzen.



In Red Hats Werkzeug system-config-cluster lassen sich unter anderem die Fencing-Techniken konfigurieren.

Kontextabhängige Pfade

Das Design der Datenaufbewahrung spielt auch bei Cluster-Dateisystemen eine wichtige Rolle. Greifen viele Rechner auf dieselben Verzeichnisse zu, können die damit verbundenen

Lock-Prozesse zu Leistungseinbußen führen. Nach Möglichkeit sollte der Anwender die Datenhaltung so strukturieren, dass diese Situation möglichst selten auftritt.

Ist das nicht möglich, beispielsweise weil die Anwendung die Konfiguration in einem bestimmten Verzeichnis sucht, hilft die Technik der kontextabhängigen Pfade. Hier greifen Anwender oder Applikation von zwei Cluster-Knoten auf das gleiche Verzeichnis zu, landen aber letzten Endes bei unterschiedlichen Datenblöcken (Listing 2). In der GFS2-Sprache heißt diese Technik Context Dependent Path Names (CDPN), während im OCFS2 von Context Dependent Symbolic Links (CDSL) die Rede ist. Als Kontext-Kriterium stehen zum Beispiel der Hostname (entspricht "uname -n"), die Maschinen-Architektur ("uname -m), das Betriebssystem ("uname -s", "uname -o") oder sogar Benutzer-Parameter ("id -u" und "id -g") zur Verfügung.

Listing 2: Kontextabhängige Pfade

```
# ssh node0 'ls /data/testrechner/'
datei.txt
# ssh node1 'ls /data/testrechner/'
andere_datei.txt
```

Volume Management im Cluster

Ohne zusätzliche Software können GFS2 und OCFS2 nicht mehrere Partitionen innerhalb eines Cluster-Dateisystems verwalten. Beide erwarten ein einziges (logisches) Block-Gerät zum Anlegen von GFS2 beziehungsweise OCFS2. Das Zusammenfassen mehrerer Partitionen erledigt der Admin mit Volume-Managemen- Software. Im Falle von GFS2 ist der CLVM (Clustered Logical Volume Manager 2) [8] das empfohlene Mittel. Bei OCFS2 kommt meistens das Enterprise Volume Management System 2 (EVMS2) von IBM zum Einsatz. Beide Anwendungen stehen unter GPL.

Der CLVM ist eine cluster-fähige Erweiterung des LVM2. Der CLVM-Daemon macht die Metadaten eines Volumes für alle Cluster-Knoten sichtbar und regelt die Zugriffe auf Block-Geräte-Ebene. Er nutzt dabei denselben Lock-Manager wie GFS und ist in Verbindung mit GFS2 der empfohlene Weg.

EVMS2 ist etwas allgemeiner als der CLVM und durch seine Plugin-Struktur einfach erweiterbar - es gibt sogar ein LVM-Plugin. Im Zusammenspiel mit Cluster-Dateisystemen nimmt EVMS2 analoge Aufgaben zum CLVM wahr. Der CLVM und das EVMS2 unterstützen das Anlegen von Snapshots, eine Funktion, die weder im GFS2 noch im OCFS2 zur Verfügung steht.

Praxis und Verfügbarkeit

Mit GFS2 und OCFS2 sind zwei Cluster-Dateisysteme im Vanilla-Kernel enthalten. Die ältere Geschichte und den längeren Praxis-Einsatz zeigt GFS2 zum Beispiel bei den Konfigurationsmöglichkeiten des Fencing und den unterstützten Lock-Modi. GFS2 kennt zudem Access Control Lists (ACLs) und Quotas. Beides ist bei OCFS2 nicht enthalten. Dafür beherrscht OCFS2 gleichzeitig den gepufferten und ungepufferten ("O_DIRECT") Zugriff.

GFS2 kann der Admin nicht so einfach testen, da vorgefertigte Software-Pakete für das clusterfähige Volume-Management und die Cluster-Software selbst nicht im Lieferumfang von Red Hats Community-Distribution Fedora enthalten sind. Das seit letztem Herbst erhältliche Unbreakable Linux von Oracle dagegen ist als Binärdistribution kostenfrei zu bekommen und enthält alle notwendigen Zutaten zur Verwendung von OCFS2. (mhu)

Der Autor

Dr. Udo Seidel ist seit 10 Jahren Linux-Fan. Er hat als Linux/Unix-Trainer, Administrator und Senior Solution Engineer gearbeitet. Seit 2006 ist er Intersystem Connectivity Specialist bei der Amadeus Data Processing GmbH in Erding