Verfügbarkeit von Applikationen in Failover Szenarien

Winfried Wojtenek

Abstract
This work is an introduction to availability and high availability of applications in a Unix environment. The basics of availability, monitoring and switching applications from one server to the failover server are described. In case of a hardware failure, multiple virtual IP addresses are used for a switch between server. Further a shared storage is necessary to get access to the data of the applications. An example for a manual fail over will be explained. Further commercial Cluster Software will be described and discussed.

1 Einleitung

Diese Einführung in die Problematik der Verfügbarkeit von Applikationen bei eventuellen Hardware Ausfällen, stellt einfache bis komplexe Lösungen vor. Zunächst wird ein einfaches manuellen Failover-Verfahren vorgestellt und der Weg bis zu einer kompletten Cluster-Lösung erörtert. Cluster- Lösungen zeichnen sich dadurch aus, dass im Fehlerfall alle Dienste des ausgefallenen Servers von einem Ersatzsystem zur Verfügung gestellt werden.

Ein spezieller Clustertyp ist der Rechen-Cluster. Beim sogenannten Load Balancing werden Aufgaben/Dienste zwischen zwei oder mehreren Servern aufgeteilt, um die vorhandene Hardware bestmöglich auszunutzen. Dieses Konzept eignet sich nicht für Failover Lösungen und wird deswegen hier nicht behandelt.

Eine Applikation ist mission critical, wenn man eine Verfügbarkeit von 24 Stunden für 7 Tage die Woche erwartet. Der folgenden Tabelle kann man die erlaubten Ausfallzeiten bei einer garantierten Verfügbarkeit pro Jahr entnehmen.

Verfügbarkeit %	Tage	Stunden	Minuten
99.000	3	16	36
99.500	1	20	48
99.900	0	9	46
99.990	0	1	53
99.999	0	0	5

Tabelle 1: System down Zeiten für ein Jahr (absolute Zeit).

Es ist eine Frage des Servicevertrags und bedarf der Definition, ob die Verfügbarkeit durch Systemwartungsarbeiten beeinträchtigt wird. Jedenfalls ist zu beachten, dass im Regelbetrieb ein Server nicht immer verfügbar ist. Je nach Betriebssystem ist das geregelte herunterfahren notwendig, beispielsweise Solaris und Patch-Cluster. Berücksichtigt man

diese geregelten Ausfälle durch Wartungsarbeiten, ist es offensichtlich, dass selbst eine $99,900\%$ Verfügbarkeit eher theoretisch ist.

2 Failover Szenarien

Prinzipiell können alle Teile der Computerhardware ausfallen. Davon sind häufiger Festplatten, Netzwerkkarten und Netzteile betroffen. Hierbei verursachte Serverausfälle sind mit Hilfe eines Backup und Recovery Systems, bei dem DLT Tapes als Speichermedien benutzt werden, nicht zeitnah zu bewältigen. Dies zeigen Rückspielzeiten der Daten von Backup Medien für grö3ere Datenbestände und Datenbanken. Für die betroffenen Applikationen sind niedrige Ausfallzeiten somit nicht gewährleistet.

Nach Hardwareausfällen werden Failover Verfahren benutzt, um die betroffenen Applikationen wieder in Betrieb zu nehmen. Vorbereitungen beinhalten eine Anschaffung entsprechender Hardware, der doppelten Netzwerkanbindung und Installation der Software. Neue Server sollten identisch mit den bereits eingesetzten Systemen sein, wobei dies Hard- und Software betrifft. Soweit möglich sollte die Hardware jedes Servers selbst redundant sein.

- doppelte Netzteile.
- Spiegeln des Betriebssystems und der Swap Disk.
- Spiegeln der Festplatten über mehrere Kontroller (RAID).
- Zusätzliche Netzwerkkarten.
- Unterschiedliche Switshe oder Hubs im Netzwerk.
- Die Applikationen und spezifische Daten der Benutzer sollten auf einem Shared Hard Disk Storage, auf das von beiden Servern zugegriffen werden kann, eingerichtet werden. Dies kann eine einzelne SCSI Hard Disk wie in Blackmon und Nguyen oder ein Fibre Channel Storage Area Netzwerk sein.

Vorbereitung des Ersatzsystems für ein Failover:

- Boot Disk spiegeln (Minimalkonfiguration).
- Dateisysteme anlegen.
- Netzwerk einrichten und IP Adressen Zuweisung durchführen.
- Virtuelle IP für die Applikationen.
- Network File System (NFS): entsprechende Verzeichnisse und mount points einrichten.
- Devices für die Datenbanken anlegen.
- Funktionstests der Netzwerkschnittstellen und des Festplatten Speicherplatzes.

Vorbereitungen für die Applikationen:

- Ausführbare Programme des Database Server und der Client Software auf den Shared Storage Festplatten einrichten.
- Gegebenenfalls Daten der Datenbanken einspielen (hot database auf einem Failover Server).

Unabhängig von der Art der Implementierung eines Failover ist die Konsistenz der Daten und Datenbanken zu gewährleisten. Die folgenden Failover Verfahren haben verschiedene technische und administrative Aufwände und bedingen unterschiedliche Ausfallzeiten.

2.1 Manuelles Failover

Ein manueller Failover von einem defekten zum Stand by Server benötigt die geringsten Eingriffe während des laufenden Betriebes. Allerdings ist der Failover mit einem hohen administrativen Aufwand verbunden. Nach Ausfall der Hardware werden Applikationen auf den Ersatzserver gestartet. Notwendige Massnahmen beinhalten:

- Die IP-Adresse und der Hostname des ausgefallenen Servers wird auf den Stand by Server übertragen.
- Mounten der Dateisysteme.
- Zuweisen von NFS Dateisystemen.
- Starten der Software.
- Starten der Datenbanken.
- Starten der Applikationen.
- Starten der Cron Jobs.

2.2 Skriptbasiertes Failover

Ein skriptbasiertes Failover wird, im besten Fall, durch den Aufruf eines einzigen Skriptes durchgeführt. In diesem Skript sind alle Arbeiten eingebettet, die bei einem manuellen Failover ausgeführt werden. Dies beinhaltet: Zuweisen der IP-Adressen, Importieren von Volume Gruppen, Mounten von Dateisystemen, das Sharen von NFS Verzeichnissen, Aktivieren von Cron Jobs und Starten der Applikationen.

Um eine Fehlerdiagnose während des laufenden Betriebes zu erleichtern, wird das Netzwerk, die Festplatten und die Aktivität der Applikationen beobachtet. Dies geschieht mit einem Checkskript, auch Heartbeat genannt. Dies kann mit einer Ping-Utility und mit einem permanenten Festplattencheck (quorum check) verwircklicht werden. Das Netzwerk wird als Fehlerquelle ausgeschlossen, indem eine zweite, private Netzwerkverbindung zwischen den Servern eingerichtet wird. Nach fehlschlagen der Checks werden die Sytemadministratoren alamiert und starten das Failover Script.

2.3 Automatisiertes Failover

Ein automatisierter Failover erfordert alle für das skriptbasiertes Failover notwendigen Massnahmen. Zusätzlich müssen die Funktionen der Primärserver mit seinen Applikationen verifizieren werden. Im Falle eines nicht verifizierbaren Zustandes wird ein Failover auf den Ersatzserver ausgelöst.

2.4 Cluster Lösungen

Für höhere Verfügbarkeit als durch automatische Failover sind Methoden einzusetzen, bei denen die Inhalte des RAM dupliziert und konsistent gehalten werden. Dies wird beispielsweise von Sybase Inc. mit dem Produkt Open Switch und von Oracle mit dem Parallel Server angeboten. Diese Lösungen sind durch Lizenskosten der Datenbankhersteller ausgesprochen kostenintensiv.

3 Kosten

Zu den Anschaffungskosten der Hardware, der Applikationen, der Clustersoftware und der Installation der Applikationen addieren sich die Arbeitskosten für den laufenden Betrieb. Ein manuelles Failover verursacht höhere innerbetriebliche Kosten und einen grösseren zeitlichen Ausfall als ein skriptbasierter Failover. Bei kommerziellen Lösungen muss zusätzlich für das Cluster, dessen Agenten und die Implementierung finanziell aufgekommen werden. Damit werden die Kosten für Clusterimplementierungen von intern auf Drittmittelkosten verschoben.

Nebst eigengebauten Lösungen, um eine höhere Verfügbarkeit zu gewährleisten, existieren Lösungen, die auf dem GNU Public Lizenz Modell basieren. Beispielsweise ist Linux-Hearbeat gut dokumentiert und günstig zu implementieren.

4 Zusammenfassung

In dieser Arbeit werden die Vorraussetzungen für die Verfügbarkeit und Hochverfügbarkeit von Applikationen erörtert. Grundlage einer Implementierung von Hochverfügbarkeitslösungen, ist eine redundant ausgelegte Hardware (Server / Cold- und Hot-Spare-Server). Dies muss bereits bei der Anschaffung der Hardware berücksichtigt werden. Ist diese technische Vorraussetzung gegeben, gibt es drei Strategien, einem Hardware-Ausfall zu begegnen, so dass die Anwender der Applikationen den Ausfall nicht bemerken. Die hier vorgestellten Strategien unterscheiden sich im wesentlichen durch ihren technischen bzw. administrativen Aufwand. Es wird folgendermassen unterschieden:
– Manuelle Verfahren, die Applikation wird per "Hand" auf einer anderen funktionierenden Hardware gestartet.
– Skriptbasierte Verfahren, d.h. die Umstellung auf eine andere Hardware wird durch vordefinierte Scripts erleichtert. Die Skripts können einen Fehler erkennen. (Halbautomatik)
– Cluster-Lösungen, meist kommerziell, funktionieren automatisch innerhalb bestimmter Zeitschranken ohne Eingriffe durch den Administrator.

Dass diese drei Strategiegruppen unterschiedliche Reaktionszeiten haben, ist offensichtlich. Dies resultiert in unterschiedlichen Fehlzeiten der Applikationen.

Literatur

A. Barak, O. La'adan: The MOSIX Multicomputer Operating System for High Performance Cluster Computing. Elsevier Science.

Steve Blackmon and John Nguyen (2001): High-Availability File Server with heartbeat. Sys Admin 2001 Vol. 10, 9.

Cluster Information von Sun Microsystems. http://www.sun.com/fullmoon

R. Greenwald, R. Stackowiak, J. Stern (2001, 2.nd): Oracle Essentials. OReilly and Associates.

A. Robertson: Linux-HA Heartbeat System Design. httpd://www.linux-ha.org/comm/HBdesign.html

S. Stringfellow, M. Klivansky and M. Barto (2000): Backup and Restore. Practices for Sun Enterprise Servers. Sun Microsystems Press.

Sun Microsystems: Sun Cluster 3.0. A white Paper. Sun Microsystems.

Sybase Inc.: Understanding High Availability. A white paper from Sybase Inc.