

공개SW 솔루션 설치 & 활용 가이드

시스템SW > 자원관리



Heartbeat



제대로 배워보자

How to Use Open Source Software

Open Source Software Installation & Application Guide



오픈소스 소프트웨어 통합지원센터
Open Source Software Support Center



CONTENTS

1. 개요
2. 기능요약
3. 실행환경
4. 설치 및 실행
5. 기능소개
6. 활용예제
7. FAQ
8. 용어정리

1. 개요



<p>소개</p>	<ul style="list-style-type: none"> • 하트 비트는 리눅스 운영체제의 고가용성을 제공하는 서비스 • Linux-HA 기본 하트 비트 하위 시스템으로 초기화 및 시스템의 시작과정지시에 스크립트 실행 • 무료 ARP를 사용하여 IP 주소 탈취도 실행되며, 2개 노드(서버) 구성에서 제대로 작동하고 더 큰 구성으로 확장 가능 		
<p>주요기능</p>	<ul style="list-style-type: none"> • 인터넷 서비스가 중간에 중단되는 일 없이 지속적으로 서비스를 할 수 있도록 하드웨어나 소프트웨어 이중화 		
<p>대분류</p>	<ul style="list-style-type: none"> • 시스템SW 	<p>소분류</p>	<ul style="list-style-type: none"> • 자원관리
<p>라이선스 형태</p>	<ul style="list-style-type: none"> • Apache license 	<p>사전설치 솔루션</p>	<ul style="list-style-type: none"> • Httpd, epel
<p>운영체제</p>	<ul style="list-style-type: none"> • Linux 	<p>버전</p>	<ul style="list-style-type: none"> • 3.0.4 (2018년 10월 기준)
<p>특징</p>	<ul style="list-style-type: none"> • UDP/IP broadcast, multicast, unicast 구현 • Bidirectional Serial Rings(더 이상 지원되지 않음) • Special ping heartbeats for routers, etc (pacemaker())로 대체됨) 		
<p>보안취약점</p>	<ul style="list-style-type: none"> • 취약점 ID : CVE-2014-0160 • 심각도 : 5.0 MEDIUM • 취약점설명 : OpenSSL 1.0.1g 이전 버전은 Heartbeat Extension 패킷을 올바르게 처리하지 못해 원격 공격자는 buffer over-read를 trigger하는 조작된 패킷으로 프로세스 메모리에서 중요한 정보를 얻을 수 있음 • 대응방안 : OpenSSL 1.0.1g 이상 업그레이드 • 참고경로 : https://www.securitytracker.com/id/1030026 		
<p>개발회사/커뮤니티</p>	<ul style="list-style-type: none"> • Linux-ha 		
<p>공식 홈페이지</p>	<ul style="list-style-type: none"> • http://linux-ha.org 		



2. 기능요약

Heartbeat

HighAvailability



- 고가용성
- Heartbeat(primary 와 standby 간에 주기적인 메시지 교환)
- Primary 고장 시 primary가 제공하던 서비스를 standby에서 제공
- Standby가 Active 될 때 특정 디바이스를 마운트 하거나 특정서비스를 시작시킬 수 있음



3. 실행환경

Heartbeat

HighAvailability



- 운영체제
 - Linux



4. 설치 및 실행

Heartbeat

HighAvailability



세부 목차

1. Linux 기본 설정
2. heartbeat 설치 및 설정



4. 설치 및 실행



4.1 Linux 기본 설정(1/5)

- Heartbeat를 설치하기 위해서 웹 서비스를 제공하는 2대의 서버 생성
- 2대의 서버에 Centos 6설치 및 Apache를 설치 후 /var/www/html 에 index.html파일을 생성한 후 IP를 호출하여 정상적으로 서비스가 되는지 확인

The image displays two terminal windows and two browser screenshots. The top-left terminal window shows the command `echo "heartbeat Test Page
Server : test001" > /var/www/html/index.html` being executed on `test001`. The bottom-left terminal window shows the same command being executed on `test002`. The top-right browser screenshot shows the page content for `http://192.168.137.8/`, displaying "heartbeat Test Page" and "Server : test001". The bottom-right browser screenshot shows the page content for `http://192.168.137.7/`, displaying "heartbeat Test Page" and "Server : test002".



4. 설치 및 실행



4.1 Linux 기본 설정(2/5)

- 2대의 서버에 호스트명 지정
 - \$ vi /etc/sysconfig/network 에서 HOSTNAME=test001
 - \$ vi /etc/sysconfig/network 에서 HOSTNAME=test002

```
root@localhost/etc/sysconfig
[root@localhost sysconfig]# cd /etc/sysconfig
[root@localhost sysconfig]# vi network

root@localhost/etc/sysconfig
NETWORKING=yes
HOSTNAME=localhost.localdomain

"network" 2L, 46C
```



4. 설치 및 실행



4.1 Linux 기본 설정(3/5)

- /etc/hosts에 server의 ip에 수정한 호스트명을 지정한 후 서버 재기동
 - \$ vi /etc/hosts 에서 192.168.137.8 test001
 - \$ vi /etc/hosts 에서 192.168.137.77 test002

```
root@localhost/etc
[root@localhost sysconfig]# cd /etc/sysconfig
[root@localhost sysconfig]# vi network
[root@localhost sysconfig]# cd /etc
[root@localhost etc]# vi hosts

root@localhost/etc
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.137.77 test002

-- INSERT --
```



4. 설치 및 실행



4.1 Linux 기본 설정(4/5)

- 호스트명 변경이 정상적으로 되었으면, 프롬프트가 [root@test001 ~]# 형태 변경
- 설치된 2개 Server 시간 동기화
 - \$ rdate -s time.bora.net

```
root@test001:~
login as: root
root@192.168.137.8's password:
Last login: Sun Nov 18 12:41:48 2018 from desktop-d09nvdk.mshome.net
[root@test001 ~]# rdate -s time.bora.net
[root@test001 ~]# █
```

```
root@test002:~
login as: root
root@192.168.137.77's password:
Last login: Sun Nov 18 12:29:22 2018 from desktop-d09nvdk.mshome.net
[root@test002 ~]# rdate -s time.bora.net
[root@test002 ~]# █
```



4. 설치 및 실행



4.1 Linux 기본 설정(5/5)

- Enterprise Linux(EPEL) 저장소 추가

- \$ rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm

```
root@test001:~
login as: root
root@192.168.137.8's password:
Last login: Sun Nov 18 12:41:48 2018 from desktop-d09nvdk.mshome.net
[root@test001 ~]# rdate -s time.bora.net
[root@test001 ~]# rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-re
lease-6-8.noarch.rpm
Retrieving http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch
.rpm
warning: /var/tmp/rpm-tmp.FOgyB6: Header V3 RSA/SHA256 Signature, key ID 0608b89
5: NOKEY
Preparing...                               ##### [100%]
 1:epel-release                             ##### [100%]
[root@test001 ~]#
[root@test001 ~]#
```

```
root@test002:~
login as: root
root@192.168.137.77's password:
Last login: Sun Nov 18 12:29:22 2018 from desktop-d09nvdk.mshome.net
[root@test002 ~]# rdate -s time.bora.net
[root@test002 ~]# rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-re
lease-6-8.noarch.rpm
Retrieving http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch
.rpm
warning: /var/tmp/rpm-tmp.KTG8DK: Header V3 RSA/SHA256 Signature, key ID 0608b89
5: NOKEY
Preparing...                               ##### [100%]
 1:epel-release                             ##### [100%]
[root@test002 ~]#
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(1/8)

- Yum으로 heartbeat 설치
- \$ yum install heartbeat*

```
root@test001:~  
Installed:  
  heartbeat.x86_64 0:3.0.4-2.el6          heartbeat-devel.x86_64 0:3.0.4-2.el6  
  heartbeat-libs.x86_64 0:3.0.4-2.el6  
Dependency Installed:  
  PyXML.x86_64 0:0.8.4-19.el6          cifs-utils.x86_64 0:4.8.1-20.el6  
  cluster-glue.x86_64 0:1.0.5-6.el6     cluster-glue-libs.x86_64 0:1.0.5-6.el6  
  keyutils.x86_64 0:1.4-5.el6          libevent.x86_64 0:1.4.13-4.el6  
  libgssglue.x86_64 0:0.1-11.el6       libtirpc.x86_64 0:0.2.1-15.el6  
  nfs-utils.x86_64 1:1.2.3-78.el6_10.1 nfs-utils-lib.x86_64 0:1.1.5-13.el6  
  perl-TimeDate.noarch 1:1.16-13.el6   resource-agents.x86_64 0:3.9.5-46.el6  
  rpcbind.x86_64 0:0.2.0-16.el6  
Complete!  
[root@test001 ~]# yum -y install heartbeat*
```

```
root@test002:~  
Installed:  
  heartbeat.x86_64 0:3.0.4-2.el6          heartbeat-devel.x86_64 0:3.0.4-2.el6  
  heartbeat-libs.x86_64 0:3.0.4-2.el6  
Dependency Installed:  
  PyXML.x86_64 0:0.8.4-19.el6          cifs-utils.x86_64 0:4.8.1-20.el6  
  cluster-glue.x86_64 0:1.0.5-6.el6     cluster-glue-libs.x86_64 0:1.0.5-6.el6  
  keyutils.x86_64 0:1.4-5.el6          libevent.x86_64 0:1.4.13-4.el6  
  libgssglue.x86_64 0:0.1-11.el6       libtirpc.x86_64 0:0.2.1-15.el6  
  nfs-utils.x86_64 1:1.2.3-78.el6_10.1 nfs-utils-lib.x86_64 0:1.1.5-13.el6  
  perl-TimeDate.noarch 1:1.16-13.el6   resource-agents.x86_64 0:3.9.5-46.el6  
  rpcbind.x86_64 0:0.2.0-16.el6  
Complete!  
[root@test002 ~]#
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(2/8)

- \$ rpm -q heartbeat -d
- 설치된 디렉토리 이동
 - \$ cd /usr/share/doc/heartbeat-3.0.4/

```
root@test001:~
Installed:
  heartbeat.x86_64 0:3.0.4-2.el6      heartbeat-devel.x86_64 0:3.0.4-2.el6
  heartbeat-libs.x86_64 0:3.0.4-2.el6

Dependency Installed:
  PyXML.x86_64 0:0.8.4-19.el6          cifs-utils.x86_64 0:4.8.1-20.el6
  cluster-glue.x86_64 0:1.0.5-6.el6     cluster-glue-libs.x86_64 0:1.0.5-6.el6
  keyutils.x86_64 0:1.4-5.el6          libevent.x86_64 0:1.4.13-4.el6
  libgssglue.x86_64 0:0.1-11.el6       libtirpc.x86_64 0:0.2.1-15.el6
  nfs-utils.x86_64 1:1.2.3-78.el6_10.1 nfs-utils-lib.x86_64 0:1.1.5-13.el6
  perl-TimeDate.noarch 1:1.16-13.el6   resource-agents.x86_64 0:3.9.5-46.el6
  rpcbind.x86_64 0:0.2.0-16.el6

Complete!
[root@test001 ~]# rpm -q heartbeat -d

root@test002:~
Complete!
[root@test002 ~]# rpm -q heartbeat -d
/usr/share/doc/heartbeat-3.0.4/AUTHORS
/usr/share/doc/heartbeat-3.0.4/COPYING
/usr/share/doc/heartbeat-3.0.4/COPYING.LGPL
/usr/share/doc/heartbeat-3.0.4/ChangeLog
/usr/share/doc/heartbeat-3.0.4/README
/usr/share/doc/heartbeat-3.0.4/apphbd.cf
/usr/share/doc/heartbeat-3.0.4/authkeys
/usr/share/doc/heartbeat-3.0.4/ha.cf
/usr/share/doc/heartbeat-3.0.4/haresources
/usr/share/man/man1/cl_status.1.gz
/usr/share/man/man1/hb_addnode.1.gz
/usr/share/man/man1/hb_delnode.1.gz
/usr/share/man/man1/hb_standby.1.gz
/usr/share/man/man1/hb_takeover.1.gz
/usr/share/man/man5/authkeys.5.gz
/usr/share/man/man5/ha.cf.5.gz
/usr/share/man/man8/apphbd.8.gz
/usr/share/man/man8/heartbeat.8.gz
[root@test002 ~]#
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(3/8)

- 설정파일 샘플 복사
 - \$ cp ha.cf haresources authkeys /etc/ha.d/
- vi를 사용하여 ha.cf 파일 수정
 - \$ vi /etc/ha.d/ha.cf

```
root@test001:/usr/share/doc/heartbeat-3.0.4
heartbeat.x86_64 0:3.0.4-2.el6      heartbeat-devel.x86_64 0:3.0.4-2.el6
heartbeat-libs.x86_64 0:3.0.4-2.el6

Dependency Installed:
PyXML.x86_64 0:0.8.4-19.el6          cifs-utils.x86_64 0:4.8.1-20.el6
cluster-glue.x86_64 0:1.0.5-6.el6    cluster-glue-libs.x86_64 0:1.0.5-6.el6
keyutils.x86_64 0:1.4-5.el6         libevent.x86_64 0:1.4.13-4.el6
libgssglue.x86_64 0:0.1-11.el6      libtirpc.x86_64 0:0.2.1-15.el6
nfs-utils.x86_64 1:1.2.3-78.el6_10.1  nfs-utils-lib.x86_64 0:1.1.5-13.el6
perl-TimeDate.noarch 1:1.16-13.el6   resource-agents.x86_64 0:3.9.5-46.el6
rpcbind.x86_64 0:0.2.0-16.el6

Complete!
[root@test001 ~]# cd /usr/share/doc/heartbeat-3.0.4/
[root@test001 heartbeat-3.0.4]# cp ha.cf haresources authkeys /etc/ha.d/
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/ha.cf
```

```
root@test002:/usr/share/doc/heartbeat-3.0.4
#
# There are lots of options in this file.  All you have to have is a set
# of nodes listed {"node ...} one of {serial, bcast, mcast, or ucast},
# and a value for "auto_failback".
#
# ATTENTION: As the configuration file is read line by line,
# THE ORDER OF DIRECTIVE MATTERS!
#
# In particular, make sure that the udpport, serial baud rate
# etc. are set before the heartbeat media are defined!
# debug and log file directives go into effect when they
# are encountered.
#
# All will be fine if you keep them ordered as in this example.
#
# Note on logging:
# If all of debugfile, logfile and logfacility are not defined,
# logging is the same as use_logd yes. In other case, they are
# respectively effective. if deterring the logging to syslog,
# /etc/ha.d/ha.cf" 340L, 10502C
```



4. 설치 및 실행



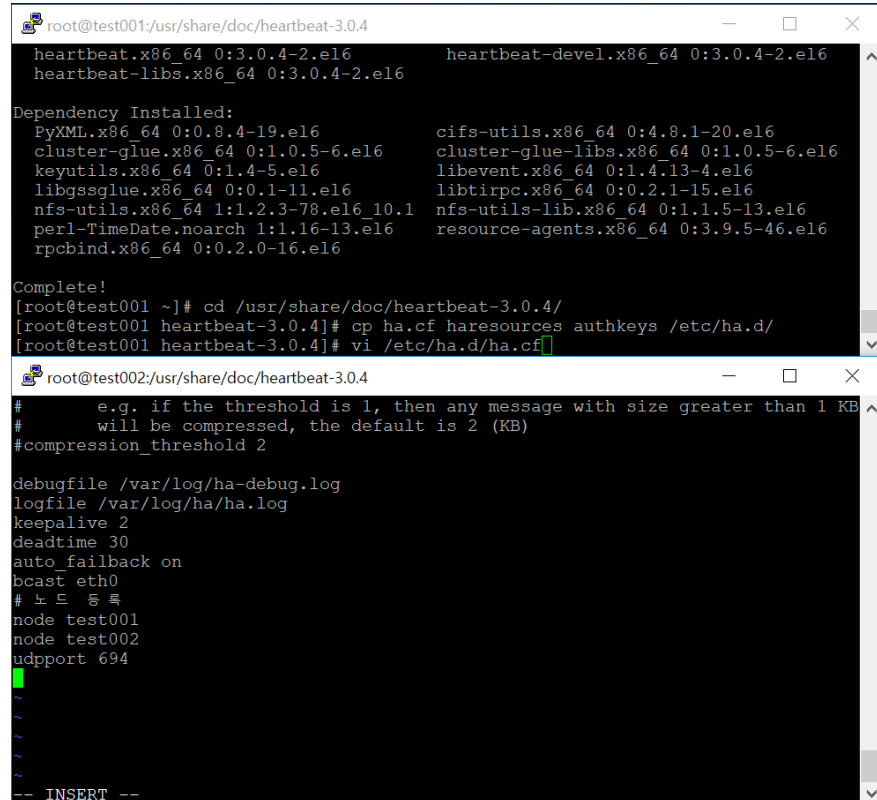
4.2 heartbeat 설치 및 설정(4/8)

- ha.cf 파일에 #으로 주석처리되어 있는 부분을 설정하면 되는데, 각 설정값의 내용은 아래와 같음

```
debugfile /var/log/ha/debug.log # 디버깅용 로그파일 위치
# logfile /var/log/ha/ha.log # 기타 로그파일 위치(사용하지 않음)
keepalive 2 # 노드간 체크 시간(두 노드간에 얼마나 자주 heartbeat를 주고받을 것인가를 설정)
deadtime 30 # 호스트가 죽었다고 판단하는 시간
auto_failback on # 자동 복구 On(한 노드가 서비스 하다가 죽었을 때 새로운 노드로 자원이 이동)
bcast eth1 # 노드간 체크에 사용할 인터페이스(heartbeat 를 보낼 인터페이스 설정)
# 노드 등록
node test001 # primary 서버의 호스트명
node test002 # standby 서버의 호스트명
udpport 694 # UDP 에 사용할 포트(통신포트 설정)
```

- 참고로 아래 내용을 ha.cf 파일 하단에 추가해 주면 기본적인 설정 마침

```
debugfile /var/log/ha/debug.log
# logfile /var/log/ha/ha.log
keepalive 2
deadtime 30
auto_failback on
bcast eth1
# 노드 등록
node test001
node test002
udpport 694
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(5/8)

- ha.cf 파일에서 지정한 로그를 위한 디렉토리 및 파일 생성
 - \$ mkdir /var/log/ha
 - \$ touch /var/log/ha/debug.log

```
root@test001:/usr/share/doc/heartbeat-3.0.4
keyutils.x86_64 0:1.4-5.el6      libevent.x86_64 0:1.4.13-4.el6
libgssglue.x86_64 0:0.1-11.el6    libtirpc.x86_64 0:0.2.1-15.el6
nfs-utils.x86_64 1:1.2.3-78.el6_10.1  nfs-utils-lib.x86_64 0:1.1.5-13.el6
perl-TimeDate.noarch 1:1.16-13.el6  resource-agents.x86_64 0:3.9.5-46.el6
rpcbind.x86_64 0:0.2.0-16.el6

Complete!
[root@test001 ~]# cd /usr/share/doc/heartbeat-3.0.4/
[root@test001 heartbeat-3.0.4]# cp ha.cf haresources authkeys /etc/ha.d/
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/ha.cf
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/debug.log
touch: cannot touch `/var/log/ha/debug.log': No such file or directory
[root@test001 heartbeat-3.0.4]# mkdir /var/log/ha
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/debug.log
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/ha.log
[root@test001 heartbeat-3.0.4]#

root@test002:/usr/share/doc/heartbeat-3.0.4
/usr/share/man/man1/ci_status.1.gz
/usr/share/man/man1/hb_addnode.1.gz
/usr/share/man/man1/hb_delnode.1.gz
/usr/share/man/man1/hb_standby.1.gz
/usr/share/man/man1/hb_takeover.1.gz
/usr/share/man/man5/authkeys.5.gz
/usr/share/man/man5/ha.cf.5.gz
/usr/share/man/man8/apphd.8.gz
/usr/share/man/man8/heartbeat.8.gz
[root@test002 ~]# cd /usr/share/doc/heartbeat-3.0.4/
[root@test002 heartbeat-3.0.4]# cp ha.cf haresources authkeys /etc/ha.d/
[root@test002 heartbeat-3.0.4]# vi /etc/ha.d/ha.cf
[root@test002 heartbeat-3.0.4]# vi /etc/ha.d/ha.cf
[root@test002 heartbeat-3.0.4]# mkdir /var/log/ha
[root@test002 heartbeat-3.0.4]# touch /var/log/ha/debug.log
[root@test002 heartbeat-3.0.4]# touch /var/log/ha/ha.log
[root@test002 heartbeat-3.0.4]#
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(6/8)

- ha.cf 파일에서 지정한 포트에 대한 방화벽 해제
 - \$ vi /etc/sysconfig/iptables
 - -A INPUT -m state --state NEW -m udp -p udp --dport 694 -j ACCEPT
 - \$ service iptables restart
 - \$ service iptables save

Ex)

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

```
root@test001:/usr/share/doc/heartbeat-3.0.4
libgssglue.x86_64 0:0.1-11.el6      libtirpc.x86_64 0:0.2.1-15.el6
nfs-utils.x86_64 1:1.2.3-78.el6_10.1  nfs-utils-lib.x86_64 0:1.1.5-13.el6
perl-TimeDate.noarch 1:1.16-13.el6      resource-agents.x86_64 0:3.9.5-46.el6
rpcbind.x86_64 0:0.2.0-16.el6

Complete!
[root@test001 ~]# cd /usr/share/doc/heartbeat-3.0.4/
[root@test001 heartbeat-3.0.4]# cp ha.cf haresources authkeys /etc/ha.d/
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/ha.cf
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/debug.log
touch: cannot touch `/var/log/ha/debug.log': No such file or directory
[root@test001 heartbeat-3.0.4]# mkdir /var/log/ha
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/debug.log
[root@test001 heartbeat-3.0.4]# touch /var/log/ha/ha.log
[root@test001 heartbeat-3.0.4]# vi /etc/sysconfig/iptables
[root@test001 heartbeat-3.0.4]#

root@test002:/usr/share/doc/heartbeat-3.0.4
# Generated by iptables-save v1.4.7 on Sun Nov 18 11:29:24 2018
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [4:528]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A INPUT -p tcp -m state --state NEW -m tcp --dport 8080 -m comment --comment "Tomcat Server port" -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 694 -j ACCEPT
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Sun Nov 18 11:29:24 2018
```



4. 설치 및 실행



4.2 heartbeat 설치 및 설정(8/8)

- authkeys(보안수준) 설정
 - \$ vi /etc/ha.d/authkeys
- 하단에 "auth2"
 - "2 sha1 비밀번호" 추가
- 보안수준 파일의 권한 변경
 - \$ chmod 600 /etc/ha.d/authkeys

```
root@test001:/usr/share/doc/heartbeat-3.0.4
[root@test001 heartbeat-3.0.4]# lokkit -p 8080:tcp
[root@test001 heartbeat-3.0.4]# vi /etc/sysconfig/iptables
[root@test001 heartbeat-3.0.4]# service iptables restart
iptables: Setting chains to policy ACCEPT: filter [ OK ]
iptables: Flushing firewall rules: [ OK ]
iptables: Unloading modules: [ OK ]
iptables: Applying firewall rules: [ OK ]
[root@test001 heartbeat-3.0.4]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[ OK ]
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/haresources
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/haresources
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/haresources
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/haresources
You have mail in /var/spool/mail/root
[root@test001 heartbeat-3.0.4]# vi /etc/ha.d/authkeys
[root@test001 heartbeat-3.0.4]#

#      sha1 is believed to be the "best", md5 next best.
#
#      crc adds no security, except from packet corruption.
#      Use only on physically secure networks.
#
#auth 1
#1 crc
#2 sha1 HI!
#3 md5 Hello!

auth 2
2 sha1 testpass

-- INSERT --
```



5. 기능소개

Heartbeat

HighAvailability



세부 목차

5.1 HA 구성

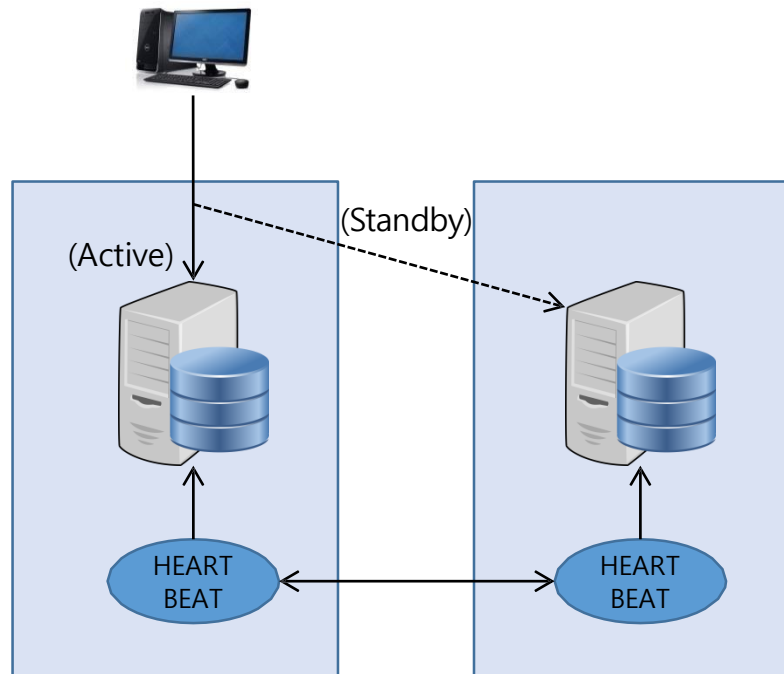


5. 기능소개



5.1 HA 구성

- 상용 IT서비스의 경우 서비스의 연속성을 지속적으로 유지하기 위하여 서버 이중화를 하게 되는데, 이때 서버끼리 서로 서버 상태에 대한 메시지를 주고 받는데 이 메시지를 HEARTBEAT라고 함
- 서버 이중화, DB서버 이중화 등 HA구성은 아래와 같은 구조로 이루어짐



6. 활용예제

Heartbeat

HighAvailability



세부 목차

6.1 Heartbeat 구성



6. 활용예제

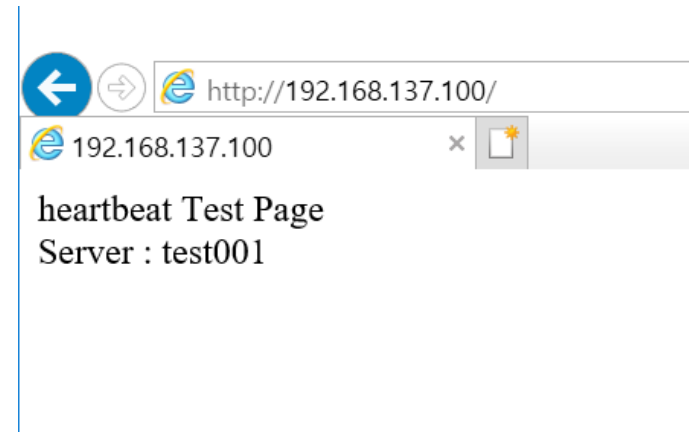
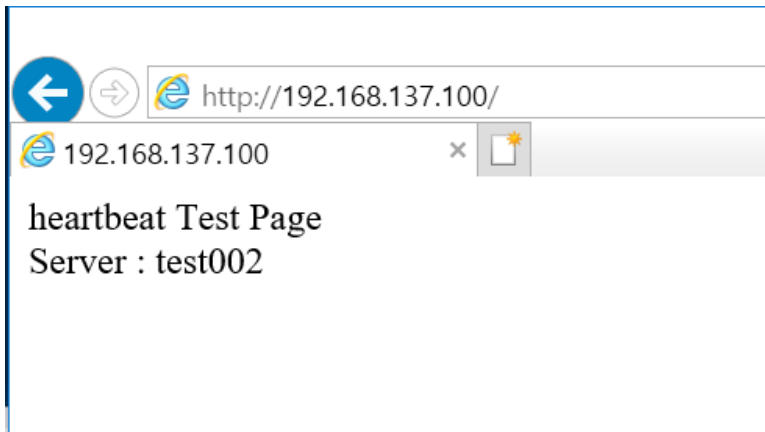


6.1 Heartbeat 구성(1/2)

- Test001 서버와 Test002 서버에서 heartbeat 서비스를 구동
 - \$ service heartbeat start
- 브라우저에서 가상 IP 192.168.137.100번으로 httpd 서비스가 정상적으로 동작되는지 확인
- Test002번 서버의 httpd서비스를 중지시키거나 Shutdown 한 후 Test001번 서버가 정상 서비스 하는지 확인
 - \$ service httpd stop
 - \$ shutdown -h now

2개 서버 모두 정상동작 중

Test002번 서버 중지 시



6. 활용예제

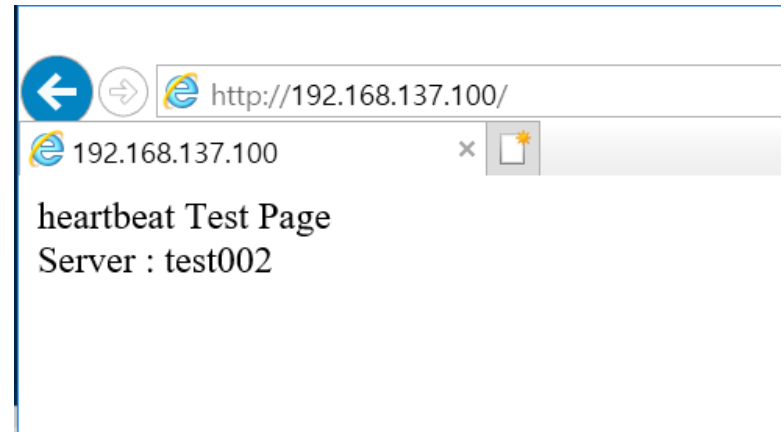
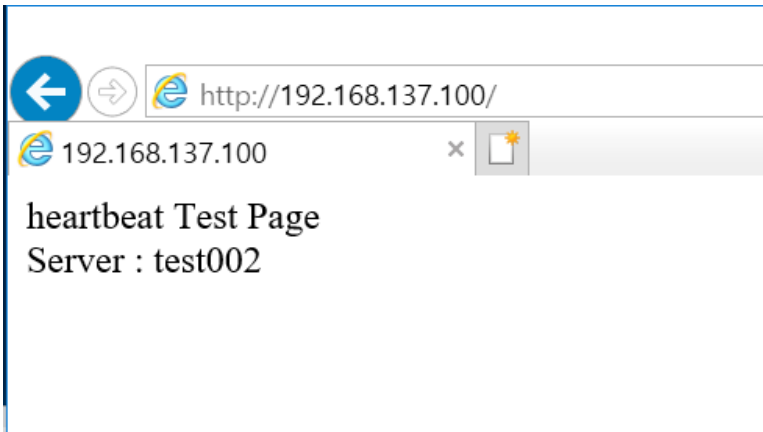


6.1 Heartbeat 구성(2/2)

- 다시 Test002번 서버의 서비스 구동
 - \$ service httpd start
- Test001번 서버의 서비스를 중지시켜 오류 없이 서비스가 정상적으로 되는지 확인
 - \$ service httpd stop

2개 서버 모두 정상동작 중

Test001번 서버 중지 시





Q HeartBeat 제약 사항이 있나요?

A 리눅스 서버상의 Fail-Over 기능을 제공하고 리눅스 운영체제에서만 사용할 수 있습니다. HeartBeat 모듈은 리눅스-HA 프로젝트 그룹에서 만든 모듈이기 때문에 리눅스 운영체제에서만 동작하는 모듈이며, 그렇기 때문에 다른 운영체제를 사용하는 시스템에서는 사용이 불가능합니다.

Q HeartBeat 구성을 위해 추가 장비가 필요한가요?

A HeartBeat 구성을 위한 추가적인 장비가 필요하지 않습니다. HeartBeat은 Active와 Passive를 사용하는 두 서버 사이의 설정으로 사용하는 모듈이기 때문에 추가적인 장비가 필요하지 않습니다.



8. 용어정리



용어	설명
고가용성(HA)	고가용성(高可用性, HA, High Availability)이란 서버와 네트워크, 프로그램 등의 정보 시스템이 상당히 오랜 기간 동안 지속적으로 정상 운영이 가능한 성질을 말한다. 고(高)가용성이란 "가용성이 높다"는 뜻으로서, "절대 고장 나지 않음" 의미
하트비트(Heartbeat)	정상 작동을 나타내거나 시스템의 일부를 동기화하는 주기 신호 또는 Linux-HA 프로젝트의 클러스터링 소프트웨어 인 Heartbeat(프로그램)
클러스터링	여러대의 컴퓨터들이 연결되어 하나의 시스템처럼 동작하는 컴퓨터들의 집합이며, 일반적으로 단일 컴퓨터보다 더 뛰어난 성능과 안정성 제공
방화벽	서로 다른 네트워크를 지나는 데이터를 허용하거나 거부하거나 검열, 수정하는 하드웨어나 소프트웨어 장치



Open Source Software Installation & Application Guide



이 저작물은 크리에이티브 커먼즈 [저작자표시-비영리-동일조건 변경허락 2.0 대한민국 라이선스]에 따라 이용하실 수 있습니다.