



# CLUSTER DE ALTA DISPONIBILIDAD

Alejandro González Pérez

*Instituto Gonzalo Nazareno*

# ¿Qué es un Cluster?

- Agrupamiento de ordenadores
- Tipos de Cluster
  - Alto rendimiento
  - Balanceo de Carga
  - Escalabilidad
  - Alta disponibilidad

# Alta Disponibilidad

→ Las tres **R** de la Alta Disponibilidad

→ *Redundancia*

→ *Redundancia*

→ *Redundancia*

# ¿Cómo funciona este cluster?

- ➔ Alta disponibilidad de Servidores
  - ➔ En caso de fallo de hardware
  - ➔ En caso de fallo de conectividad
- ➔ Alta disponibilidad de Conexión a Internet
  - ➔ En caso de fallo de router
  - ➔ En caso de fallo de conexión

# Hardware utilizado

→ Routers LINKSYS

→ Servidores

→ Equipos antiguos

→ No es necesario el mismo hardware

→ Utilizado dos HD por equipo



# Software Utilizado

- S.O LINUX

- Debian Etch

- Software

- Apache2

- openssh-server

- Heartbeat

- Script propios

- DRBD

# Configuración de red

→ Corman:

→ Interfaz eth0(principal)

★ IP:192.168.2.5

→ Interfaz eth2

★ IP:172.168.0.1

→ Rabiús:

→ Interfaz eth0(principal)

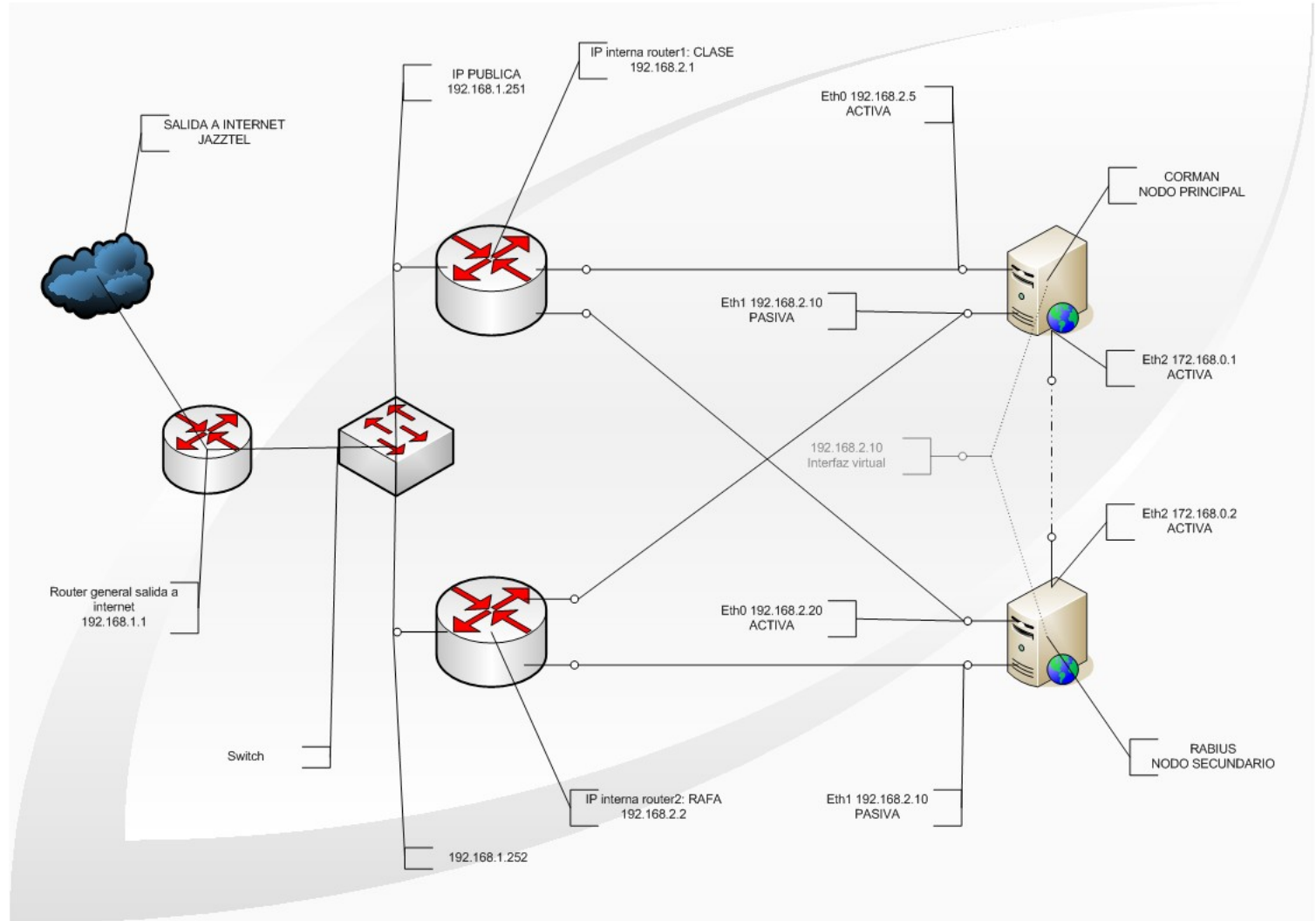
★ IP:192.168.2.20

→ Interfaz eth2

★ IP:172.168.0.2

→ Ambas interfaces ETH1 desconectadas

# Esquema de red

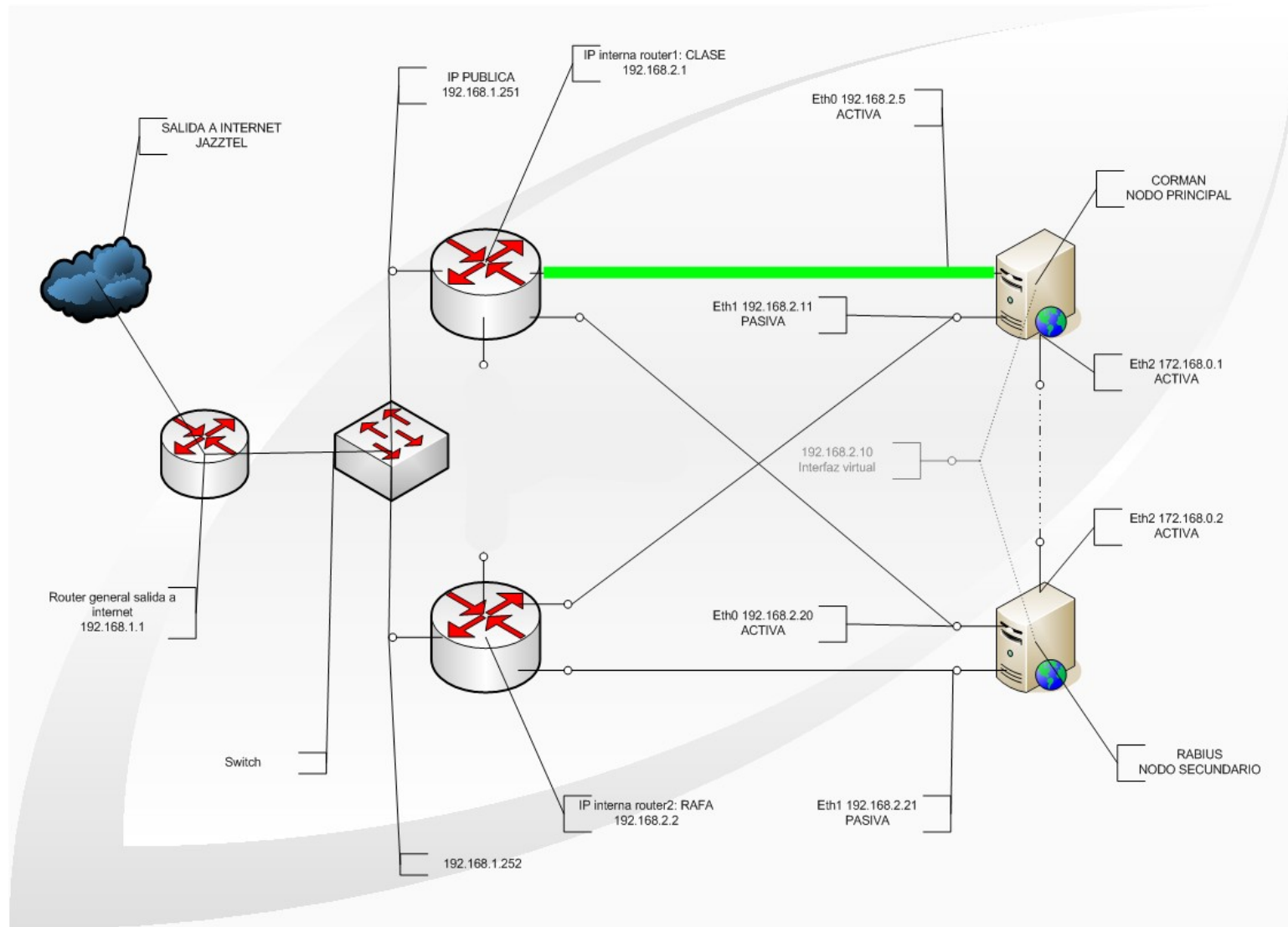




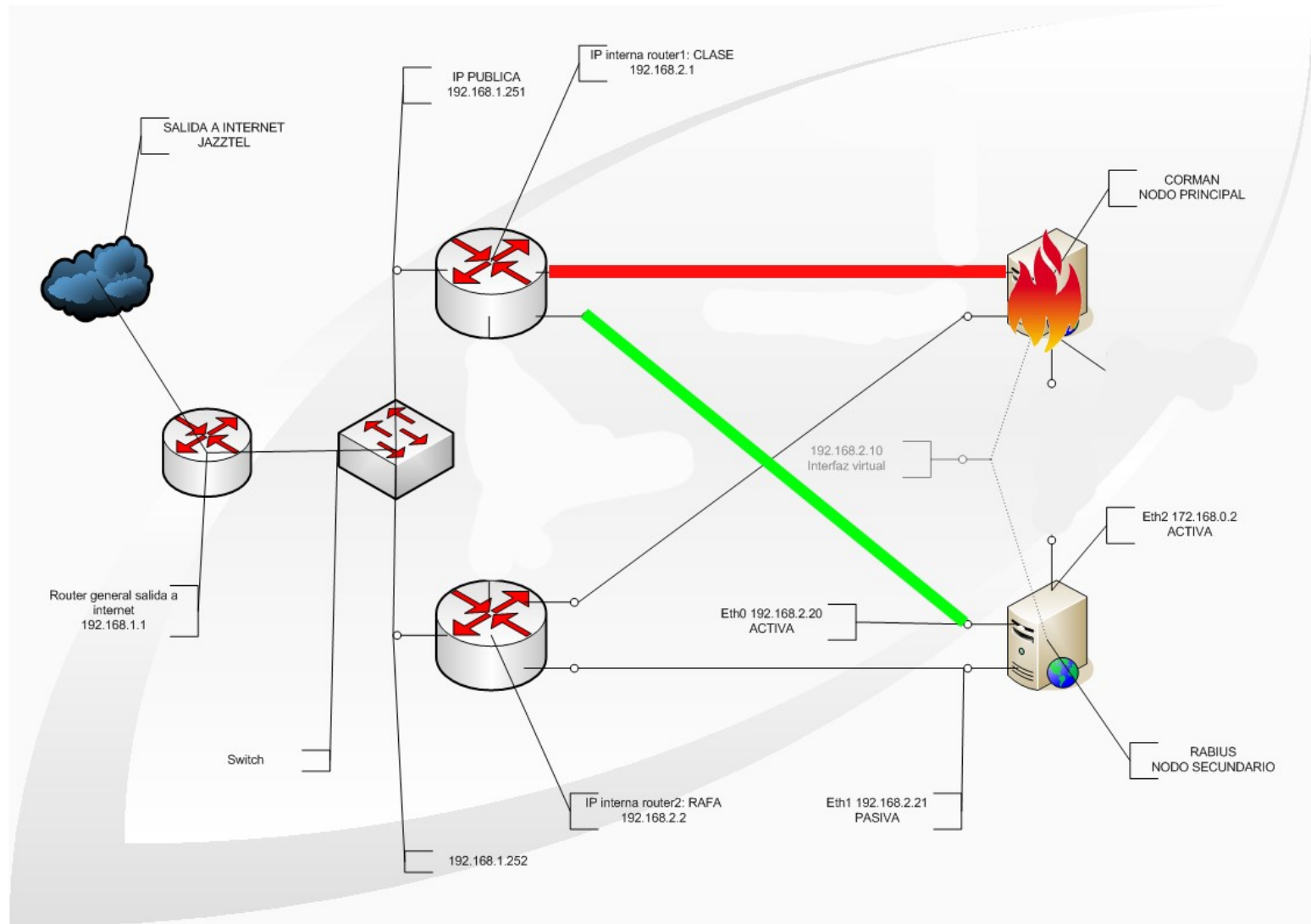
# Más detalles...

- Configuración de routers
  - IP estática.
  - Conexión entre redes 192.168.1.0 y 192.168.2.0.
  - NAT para 192.168.2.10
- Configuración básica de Apache
- Tres interfaces con la misma IP

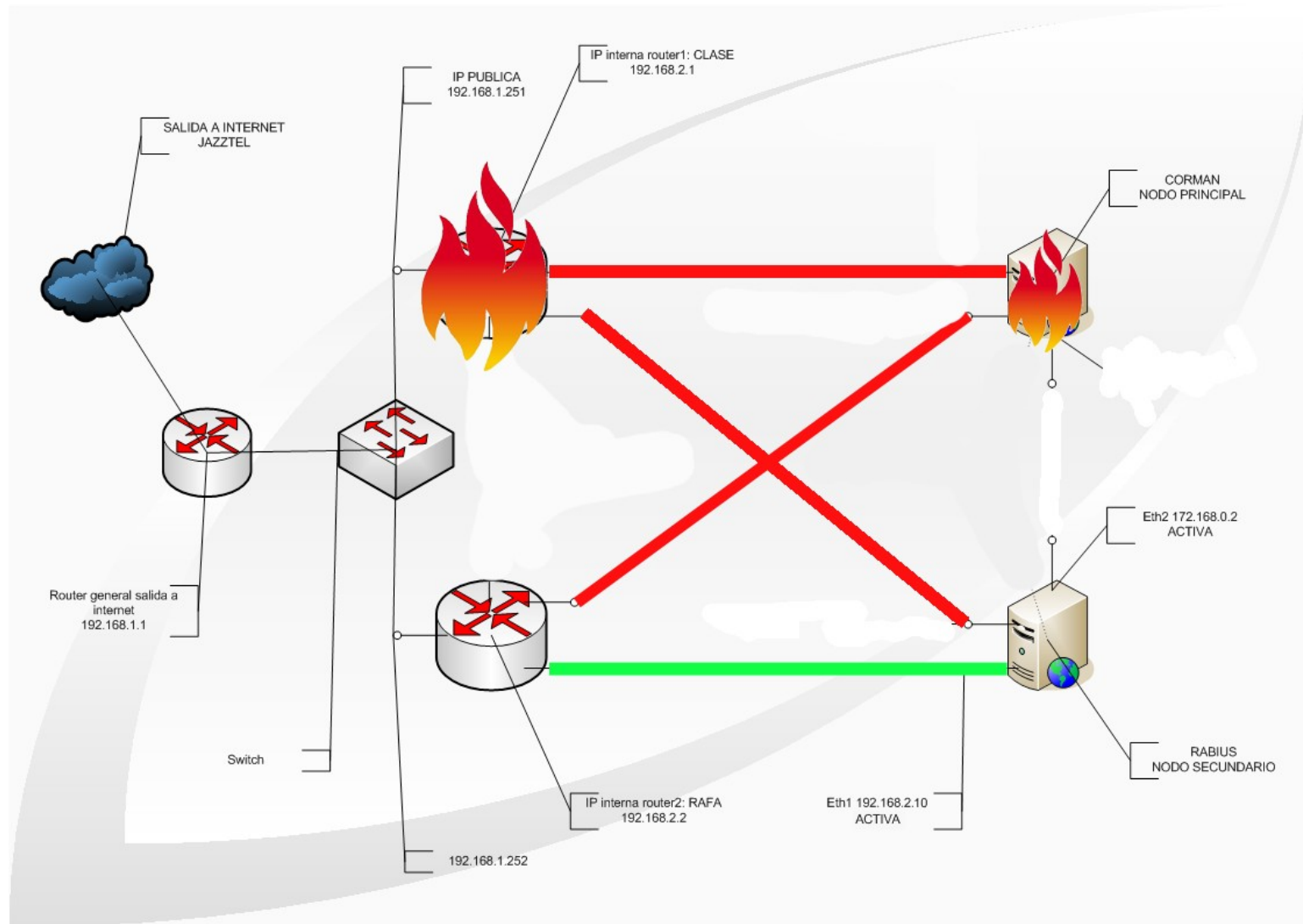
# Funcionamiento Normal



# Funcionamiento Fallo Corman



# Funcionamiento Fallo Router1





# Heartbeat

- Significa *Latido del Corazón*
- Software encargado del cluster
- Cada equipo del sistema se llama Nodo
- Comunicación constante entre nodos
- Archivos de configuración

# Archivos de configuración

- Estan en /etc/ha.d/
- Son tres:
  - authkeys
  - haresources
  - ha.cf
- Idénticos en los dos servidores

# authkeys

- Se define la clave de comunicación
- Si falla, la comunicación no existe.
- El código es:
  - ★ auth 1
    - 1 sha clavesupersecreta
    - 2 md5 clavemássecretaaún

# hairesources

- Contiene los recursos a compartir
  - Un recurso es todo lo que es manejado por el cluster que no es un nodo.
  - Los recursos deben estar en `/etc/ha.d/resource.d`
- La línea es:
  - ★ `Corman Ipaddr2::192.168.2.10/24/eth0:0 apache2`



# ha.cf

- Es el archivo principal de configuración
- Las variables más importantes son:
  - Nodos del cluster
  - Tiempo entre latidos
  - Interfaces por las que funciona el latido
  - Devolver o no el control al nodo primario

# ¿Qué conseguimos con esto?

- Alta disponibilidad
- Respaldo de servicio web
- Respaldo de red para la IP  
192.168.2.10



# Scripts

- Problema a solucionar
- Cinco scripts necesarios
  - pingcheck
  - failcase
  - Newdns
  - checkoldping
  - Backnormal
- Todos creados en `/etc/ha.d/resource.d/`
- Generan logs

# pingcheck

- Encargado de chequear internet
- Lanzamiento de failcase
- Este script estará en /usr/sbin

# failcase

- Desconecta eth0
- Levanta eth1 con IP 192.168.2.10
- Añade puerta de enlace 192.168.2.2 para eth1
- Lanza newdns y checkoldping



# newdns

- Utilizando NO-IP
- Dos configuraciones diferentes, una por cada interfaz
- Utiliza parámetros
- Recarga el script de no-ip

# checkoldping

- ➔ Se encarga de comprobar la antigua puerta de enlace
- ➔ El método es chequear la IP externa del router 1
- ➔ Si el router 1 vuelve a funcionar, lanza backnormal

# backnormal

- Se lanza al volver cuando vuelve a funcionar el router 1
- Desconecta la eth1 y reinicia los servicios de red
- Vuelve a levantar pingcheck para comprobar si falla de nuevo router1
- Lanza newdns con parámetro nuevo



# Automatización de Scripts

- Creación de nuevo script para control del demonio en `/etc/init.d/`
- Añadirla al arranque del servidor principal Corman, pero no de Rabiús
- Añadirla al archivo *haresources*

# ¿Qué conseguimos con esto?

- Alta disponibilidad de conexión a internet
- Archivo de log de todo lo relacionado con las conexiones
- Automatización de las tareas, ofreciendo respaldo de conexión



# DRBD

- Es un módulo para el kernel
- Sistema distribuido de almacenamiento
- Necesita una partición en cada servidor

# DRBD

- Actúa en modo primario/secundario
- Se configura mediante `/etc/drbd`
  - Este archivo es idéntico en ambos servidores
  - Definimos sincronización por tiempo y megas
- Se monta en `/data/`
- Se añade a *haresources*:

```
corman IPaddr2::192.168.2.10/24/eth0:0 apache2 pingcheck  
drbddisk::data Filesystem::/dev/drbd0::/data::ext3
```

# DRBD; Utilidad real

- ➔ Sirve para tener un volumen con los mismos datos en los dos servidores
- ➔ Utilizamos esto para montar un virtual host de apache
- ➔ Montamos una página web en /data/



# Y ahora..

- ➔ Comprobemos que funciona realmente.