

# Présentation de LibrA-LVS

→ Haute-disponibilité &  
Répartition de charge



LibrA-LVS

- ▶ Intégration de **logiciels libres**
  - Kernel Linux
  - IPVS (module de Netfilter depuis kernel 2.6.10)
  - Keepalived
- ▶ Technologie **Layer-4 Switching** (TCP / UDP) du kernel Linux
- ▶ Apporte à vos applications et services critiques, **haute-disponibilité** et **répartition de charge**
- ▶ Largement éprouvé en environnement de production (depuis 2000)

# LibrA-LVS : Pourquoi ?

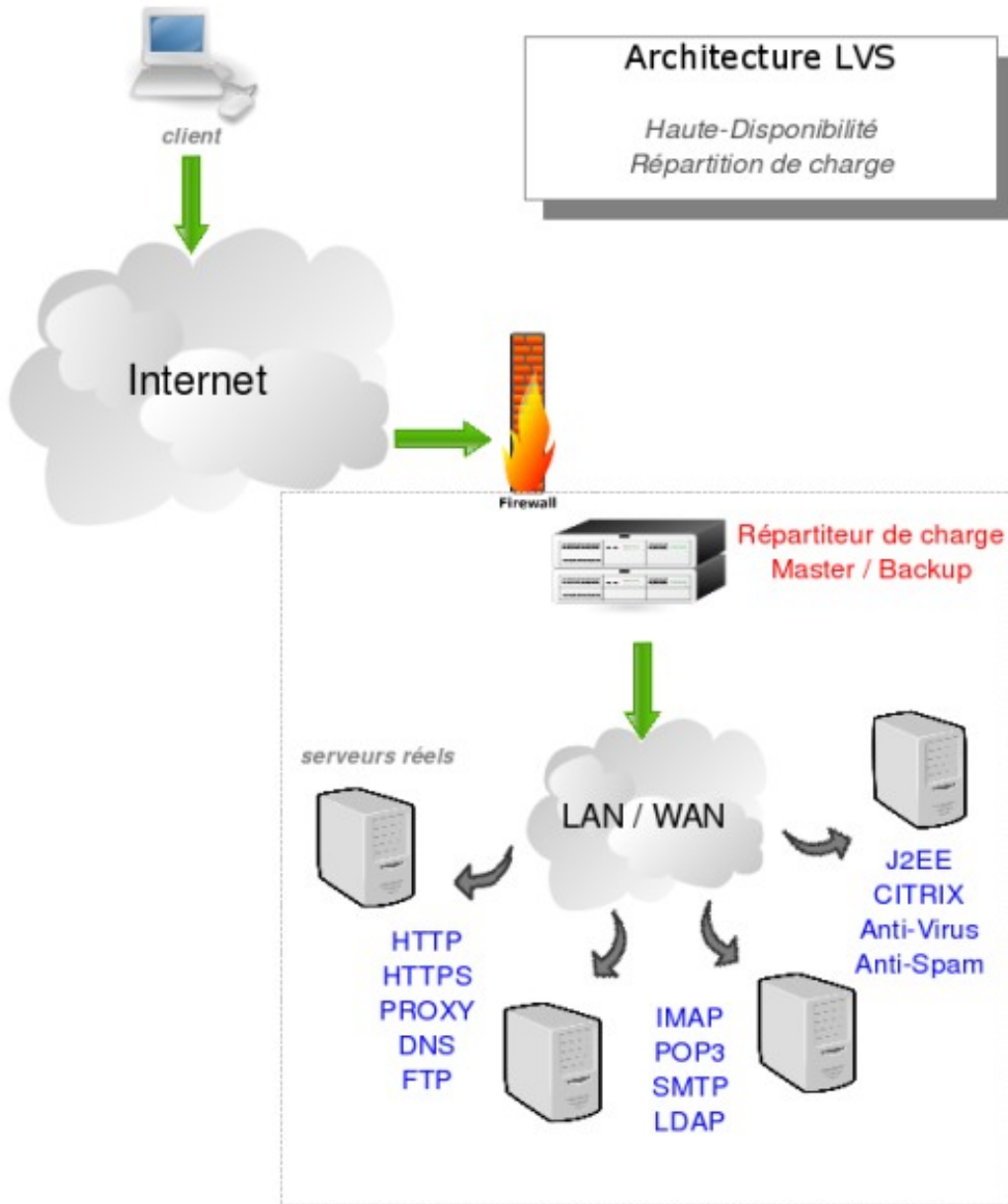
## ▶ Le Besoin

- Haute disponibilité d'un service critique
- et/ou Montée en charge

## ▶ Les services typiques qui se prêtent à un équilibrage de charge

- Mail : IMAP, POP, SMTP
- Proxy cache et reverse-proxy
- Web : HTTP, HTTPS
- J2EE : Websphere, WebLogic, JBOSS, Tomcat
- FTP, DNS, LDAP
- Samba, NFS
- Citrix MetaFrame
- Serveurs d'impression
- Filtrage de flux : Anti-Spam, Anti-Virus
- ...

# LibrA-LVS : Architecture



- ▶ Notion de **service virtuel** (VIP)
- ▶ Cluster LVS en **Failover VRRP** (Linux)
- ▶ Répartition de charge des sessions réseaux TCP/IP - NAT
- ▶ Ferme de **n serveurs réels** Linux, Unix, Windows ..
- ▶ **Services réels** hébergés par les serveurs réels (RIP)
- ▶ **Keepalive** des services réels Framework TCP Checker L3-L4-L5
- ▶ Plusieurs implémentations LVS : LVS-NAT, LVS-DR, IP Tunneling

# LibrA-LVS : Fonctionnalités

- ▶ **Répartition de charge** des sessions réseaux TCP/IP – L3 & L4
- ▶ Translation d'adresse/port (**NAT**) – **OS client indépendant**  
ou Direct Routing / IP-Tunneling
- ▶ x **services virtuels (VIP)** associés à y serveurs réels (RIP)
- ▶ Choix de l'**algorithme** de répartition de charge adapté à chaque service virtuel (VIP)
- ▶ **Persistance** des connexions clientes (TCP / UDP)
- ▶ **Synchronisation** table des connexions entre les 2 LVS
- ▶ **Maintient des connexions en cours** sans interruption de service
- ▶ **Synchronisation des instances VIP** entre les 2 LVS
- ▶ **Keepalive** et surveillance des services réels
- ▶ **Haute-disponibilité** globale des services virtuels
- ▶ Modes **actif / passif** ou **actif / actif**
- ▶ **Failover VRRP** du cluster LVS

# LibrA-LVS : Les algorithmes

## ► Les algorithmes de répartition de charge au choix :

- Round-Robin : barillet tournant, le plus simple
- Round Robin pondéré : barillet tournant avec une pondération sur les éléments de la liste. La pondération peut, par exemple, refléter la capacité de charge des serveurs réels
- Least Connection : session attribuée au serveur réel qui a le moins de connexions au moment du choix
- Least Connection pondéré : idem avec une pondération
- Destination Hashing Scheduling
- Source Hashing Scheduling
- Never Queue Scheduling (idle server or minimum expected delay)
- ..

# LibrA-LVS : Keepalive des services

- ▶ LibrA-LVS embarque un mécanisme ([keepalived](#)) permettant de **surveiller** à intervalles réguliers, **la bonne santé de tous les services des serveurs réels** :
  - ICMP Check
  - UDP Check
  - TCP Check
  - HTTP GET , SSL GET (md5 checksum)
  - MISC CHECK : checker spécifique TCP (ldap par exemple)
- ▶ **Prise de décision** et **configuration dynamique** du LVS par :
  - Ajout / retrait de serveurs réels associé à une VIP
  - Appels systèmes (programme externe, notification e-mails..)
  - Redirection vers un serveur d'excuse si tous les serveurs réels associés à un service virtuel sont indisponibles
  - log des messages via syslogd

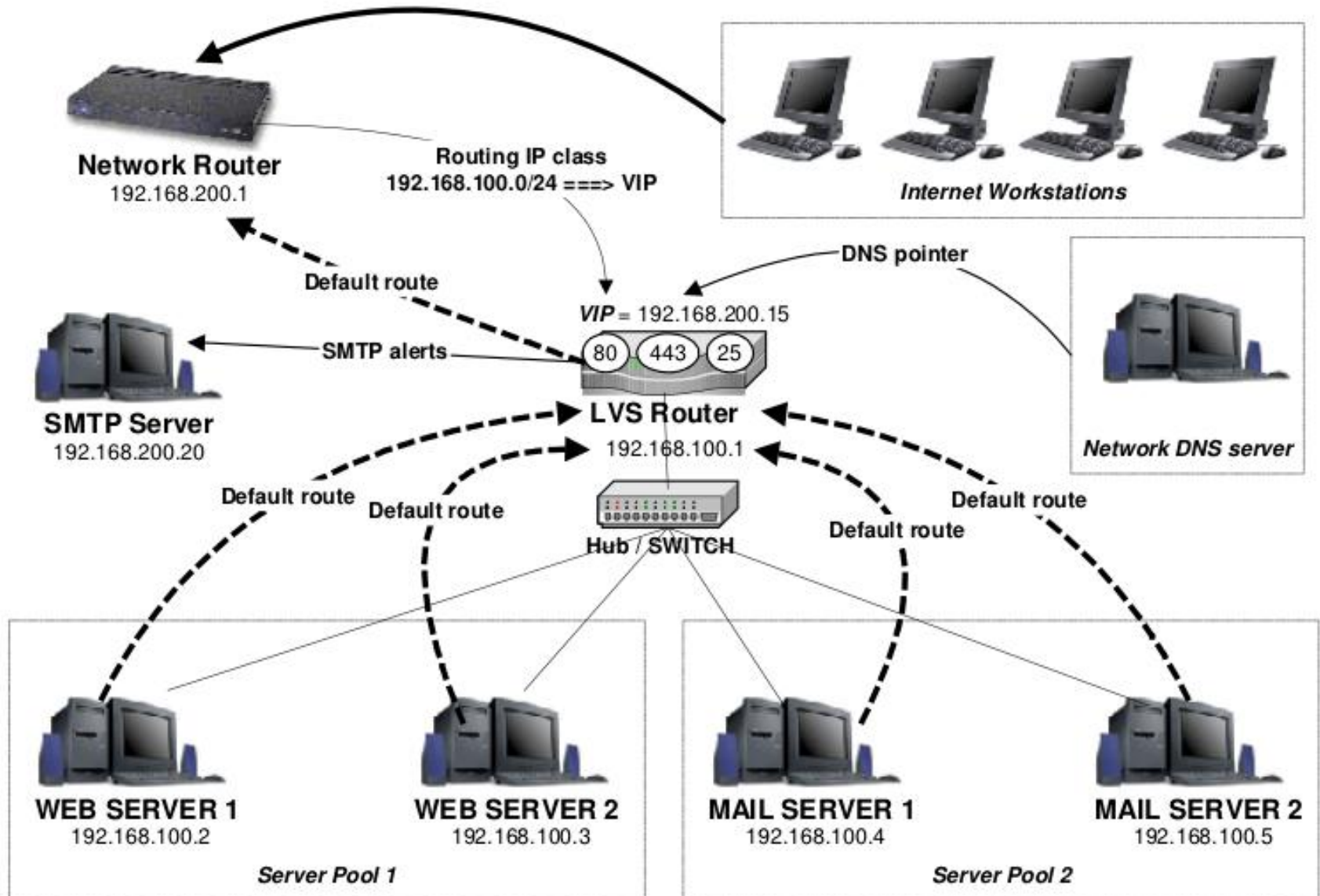
# LibrA-LVS : Failover du cluster LVS

- Pour organiser la **synchronisation des tables de routage** LVS entre les 2 serveurs frontaux du cluster LVS, et **gérer les bascules d'état maître / esclave**, LibrA-LVS embarque un mécanisme qui répond à ces besoins et intègre :
- une "**machine à états finis**" pour gérer les bascules d'état
  - le pilotage de la synchronisation des tables de routage LVS
  - un **battement de coeur** multicast, **crypté SSL** (principe du protocole VRRPv2 de Cisco) pour surveiller chaque noeud du cluster LVS
  - **Synchronisation des instances VIP** lors de bascules d'état
  - la logique d'annonce ARP de reprise d'adresse IP aux clients pour un "**IP-takeover**" **sous la seconde**
  - possibilité de n serveurs frontaux en VRRPv2, multi-site, actif-actif

# LibrA-LVS : Points forts

- ▶ **Reprise** des connexions clientes en cours, si bascule
- ▶ **Rapidité** des bascules LVS
- ▶ **Montée en charge** facile
- ▶ **Souplesse** d'administration et d'exploitation des serveurs
- ▶ **Persistance** des connexions clientes (Session ID, https..)
- ▶ **Keepalive** des serveurs réels et services associés
- ▶ **Transparent** pour les clients
- ▶ Overhead CPU et latence réseau minimale
- ▶ **Moindre coût** vs solutions hardware (Cisco ArrowPoint, F5, Dell..) - matériel : 2 machines x86 de gamme standard
- ▶ Fonctionne comme une **boîte noire**
- ▶ **Mutualisation** du LVS
- ▶ **Performance** et **Fiabilité** extrême

# LibrA-LVS : Exemple





*"La libre expression de  
votre système d'information"*

Tony OGER  
<[tony.oger@libra-linux.com](mailto:tony.oger@libra-linux.com)>