Red Hat Enterprise Linux 5

Virtual Server Administration

Linux Virtual Server (LVS) for Red Hat Enterprise Linux

Edizione 5

Last Updated: 2017-10-16

Red Hat Enterprise Linux 5 Virtual Server Administration

Linux Virtual Server (LVS) for Red Hat Enterprise Linux
Edizione 5

Landmann
rlandmann@redhat.com

Nota Legale

Sommario

Building a Linux Virtual Server (LVS) system offers highly-available and scalable solution for production services using specialized routing and load-balancing techniques configured through the PIRANHA. This book discusses the configuration of high-performance systems and services with Red Hat Enterprise Linux and LVS for Red Hat Enterprise Linux 5.

Indice

INTRODUCTION

This document provides information about installing, configuring, and managing Red Hat Virtual Linux Server (LVS) components. LVS provides load balancing through specialized routing techniques that dispatch traffic to a pool of servers. This document does not include information about installing, configuring, and managing Red Hat Cluster software. Information about that is in a separate document.

The audience of this document should have advanced working knowledge of Red Hat Enterprise Linux and understand the concepts of clusters, storage, and server computing.

This document is organized as follows:

- Capitolo 1, *Panoramica del Linux Virtual Server*
- Capitolo 2, *Configurazione LVS iniziale*
- Capitolo 3, *Impostazione del LVS*
- Capitolo 4, *Configurazione dei router LVS con**Piranha Configuration Tool***
- Appendice A, *Utilizzo di LVS con il Red Hat Cluster*

For more information about Red Hat Enterprise Linux 5, refer to the following resources:

- *Red Hat Enterprise Linux Installation Guide*— Provides information regarding installation of Red Hat Enterprise Linux 5.
- *Red Hat Enterprise Linux Deployment Guide*— Provides information regarding the deployment, configuration and administration of Red Hat Enterprise Linux 5.

For more information about Red Hat Cluster Suite for Red Hat Enterprise Linux 5, refer to the following resources:

- *Red Hat Cluster Suite Overview*— Provides a high level overview of the Red Hat Cluster Suite.
- *Configuring and Managing a Red Hat Cluster*— Provides information about installing, configuring and managing Red Hat Cluster components.
- *Logical Volume Manager Administration*— Provides a description of the Logical Volume Manager (LVM), including information on running LVM in a clustered environment.
- *Global File System: Configuration and Administration*— Provides information about installing, configuring, and maintaining Red Hat GFS (Red Hat Global File System).
- *Global File System 2: Configuration and Administration*— Provides information about installing, configuring, and maintaining Red Hat GFS2 (Red Hat Global File System 2).
- *Using Device-Mapper Multipath*— Provides information about using the Device-Mapper Multipath feature of Red Hat Enterprise Linux 5.
- *Using GNBD with Global File System*— Provides an overview on using Global Network Block Device (GNBD) with Red Hat GFS.

- *Red Hat Cluster Suite Release Notes*— Provides information about the current release of Red Hat Cluster Suite.

Red Hat Cluster Suite documentation and other Red Hat documents are available in HTML, PDF, and RPM versions on the Red Hat Enterprise Linux Documentation CD and online at http://www.redhat.com/docs/.

1. FEEDBACK

If you spot a typo, or if you have thought of a way to make this manual better, we would love to hear from you. Please submit a report in Bugzilla (http://bugzilla.redhat.com/bugzilla/) against the component **Documentation-cluster**.

Be sure to mention the manual's identifier:

```
Virtual_Server_Administration(EN)-5 (2010-02-08T16:55)
```

By mentioning this manual's identifier, we know exactly which version of the guide you have.

If you have a suggestion for improving the documentation, try to be as specific as possible. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

CAPITOLO 1. PANORAMICA DEL LINUX VIRTUAL SERVER

Il Linux Virtual Server (LVS) è composto da un set di componenti software per il bilanciamento del carico IP attraverso un insieme di real server. Viene eseguito su di una coppia di computer configurati in modo identico: uno come *router LVS attivo*, e l'altro come *router LVS di backup*. Il router LVS attivo serve per:

- Bilanciare il carico attraverso i real server.
- Per controllare l'integrità dei servizi su ogni real server.

Il router LVS di backup controlla il router LVS attivo, e ne assume il controllo in caso di un suo fallimento.

Questo capitolo fornisce una panoramica dei componenti LVS e delle loro funzioni, e consiste nelle seguenti sezioni:

- Sezione 1.1, «A Basic LVS Configuration»
- Sezione 1.2, «A Three-Tier LVS Configuration»
- Sezione 1.3, «Panoramica sulla schedulazione LVS»
- Sezione 1.4, «Metodi d'instradamento»
- Sezione 1.5, «Persistenza e Firewall mark»
- Sezione 1.6, «LVS — Un diagramma a blocchi»

1.1. A BASIC LVS CONFIGURATION

Figura 1.1, «A Basic LVS Configuration» shows a simple LVS configuration consisting of two layers. On the first layer are two LVS routers — one active and one backup. Each of the LVS routers has two network interfaces, one interface on the Internet and one on the private network, enabling them to regulate traffic between the two networks. For this example the active router is using *Network Address Translation* or *NAT* to direct traffic from the Internet to a variable number of real servers on the second layer, which in turn provide the necessary services. Therefore, the real servers in this example are connected to a dedicated private network segment and pass all public traffic back and forth through the active LVS router. To the outside world, the servers appears as one entity.

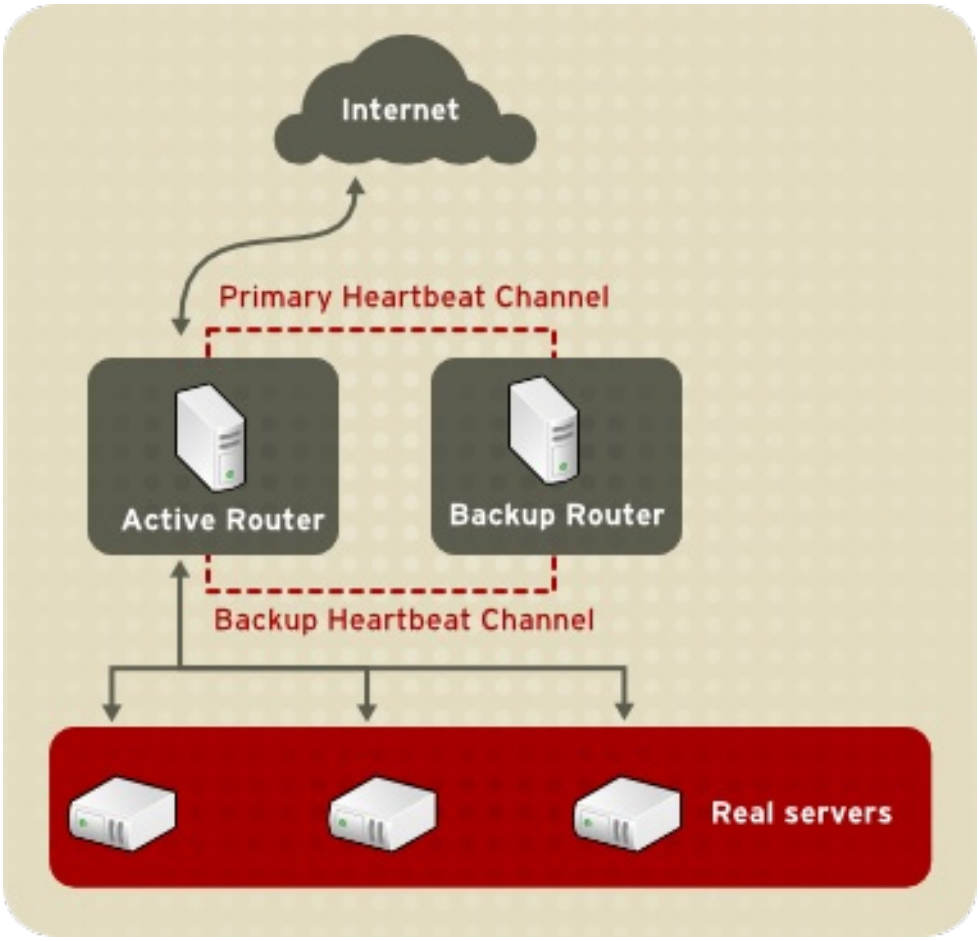


Figura 1.1. A Basic LVS Configuration

Service requests arriving at the LVS routers are addressed to a *virtual IP* address, or *VIP*. This is a publicly-routable address the administrator of the site associates with a fully-qualified domain name, such as www.example.com, and is assigned to one or more *virtual servers*. A virtual server is a service configured to listen on a specific virtual IP. Refer to Sezione 4.6, «VIRTUAL SERVERS» for more information on configuring a virtual server using the **Piranha Configuration Tool**. A VIP address migrates from one LVS router to the other during a failover, thus maintaining a presence at that IP address (also known as *floating IP addresses*).

È possibile eseguire l'alias degli indirizzi VIP sullo stesso dispositivo che collega il router LVS ad internet. Per esempio, se eth0 è collegato ad internet, allora sarà possibile eseguire l'alias dei server virtuali multipli su `eth0:1`. Alternativamente ogni server virtuale può essere associato con un dispositivo separato per servizio. Per esempio, il traffico HTTP può essere gestito su `eth0:1`, ed il traffico FTP gestito su `eth0:2`.

Only one LVS router is active at a time. The role of the active router is to redirect service requests from virtual IP addresses to the real servers. The redirection is based on one of eight supported load-balancing algorithms described further in Sezione 1.3, «Panoramica sulla schedulazione LVS».

Il router dinamico controlla dinamicamente anche lo stato generale dei servizi specifici sui real server, attraverso *script send/expect* semplici. Per un ausilio al rilevamento dello stato dei servizi che necessitano di dati dinamici, come ad esempio HTTP o SSL, l'amministratore potrà chiamare anche eseguibili esterni. Se un servizio su di un real server funziona incorrettamente, il router attivo sospende l'invio dei lavori al server in questione, fino a quando non verrà ripristinato un funzionamento normale.

Il router di backup esegue il ruolo di un sistema in standby. Periodicamente i router LVS scambiano messaggi heartbeat attraverso l'interfaccia pubblica esterna primaria e, in condizioni di failover,

attraverso l'interfaccia privata. Se il nodo di backup non riceve un messaggio heartbeat entro un intervallo di tempo previsto, esso inizierà un processo di failover, assumendo il ruolo del router attivo. Durante il processo di failover, il router di backup assume il controllo degli indirizzi VIP serviti dal router fallito, utilizzando una tecnica conosciuta come *ARP spoofing* — dove il router LVS di backup si presenta come destinazione per i pacchetti IP inviati al nodo fallito. Quando il suddetto nodo ripristina il proprio servizio attivo, il nodo di backup assume il suo ruolo di hot-backup.

The simple, two-layered configuration used in Figura 1.1, «A Basic LVS Configuration» is best for serving data which does not change very frequently — such as static webpages — because the individual real servers do not automatically sync data between each node.

1.1.1. Riproduzione dati e loro condivisione tra real server

Poichè non è presente alcun componente interno a LVS per poter condividere gli stessi dati tra i real server, l'amministratore è in possesso di due opzioni:

- Sincronizzare i dati attraverso il gruppo di real server
- Aggiungere un terzo livello alla topologia per un accesso dei dati condivisi

È preferibile la prima opzione per quei server che non permettono ad un gran numero di utenti di caricare o modificare i dati sui real server. Se la configurazione permette invece ad un gran numero di utenti di modificare i dati, come ad esempio un sito web e-commerce, allora sarà preferibile aggiungere un terzo livello.

1.1.1.1. Configurazione dei real server per sincronizzare i dati

Per un amministratore sono diponibili diversi metodi per poter sincronizzare i dati in un gruppo di real server. Per esempio, è possibile implementare gli script della shell in modo tale che se un Web engineer aggiorna un apagina, la pagina stessa viene inviata simultaneamente su tutti i server. Altresì, l'amministratore del sistema può utilizzare programmi come **rsync**, per riprodurre i dati modificati attraverso tutti i nodi ad un intervallo di tempo prestabilito.

Tuttavia questo tipo di sincronizzazione dei dati non funziona in maniera ottimale, se la configurazione è sovraccarica con utenti che caricano costantemente file o che emmetono transazioni del database. Per una configurazione con un carico molto elevato, una *topologia a tre livelli* risulta essere la soluzione ideale.

1.2. A THREE-TIER LVS CONFIGURATION

Figura 1.2, «A Three-Tier LVS Configuration» shows a typical three-tier LVS topology. In this example, the active LVS router routes the requests from the Internet to the pool of real servers. Each of the real servers then accesses a shared data source over the network.

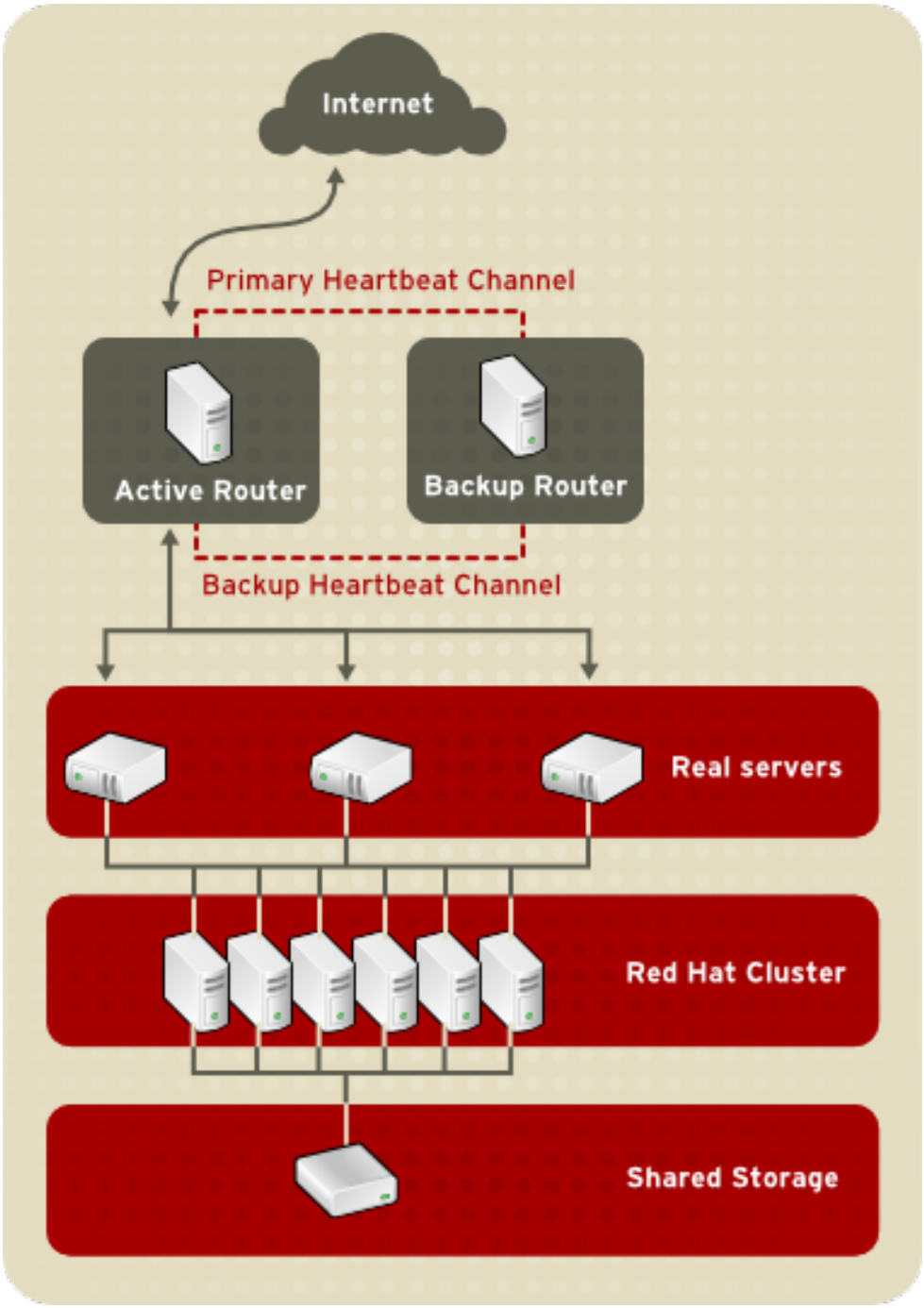


Figura 1.2. A Three-Tier LVS Configuration

Questa configurazione è ideale per server FTP molto occupati, dove i dati accessibili vengono conservati in un server 'highly available' centrale, e disponibili ad ogni real server tramite una directory NFS esportata o condivisione Samba. Questa topologia è consigliata per siti web che accedono ad un database 'highly available' centrale per le loro transazioni. Altresì, utilizzando una configurazione active-active con Red Hat Cluster Manager, gli amministratori possono configurare un cluster high-availability, per servire entrambi i ruoli simultaneamente.

Il terzo livello negli esempi sopra riportati, non è obbligato ad utilizzare Red Hat Cluster Manager, ma se non utilizza una soluzione highly available, causerà un single point of failure critico.

1.3. PANORAMICA SULLA SCHEDULAZIONE LVS

Uno dei vantaggi dell'utilizzo di LVS è rappresentato dalla sua capacità di eseguire un bilanciamento del carico a livello-IP flessibile nel gruppo dei real server. Questa flessibilità è dovuta alla varietà di

algoritmi di schedulazione, che un amministratore può scegliere quando configura LVS. Il bilanciamento del carico LVS risulta essere superiore rispetto a metodi meno flessibili, come ad esempio *Round-Robin DNS*, dove la natura gerarchica di DNS ed il caching da parte di macchine client, può portare ad uno squilibrio del carico. In aggiunta, un livello basso di filtraggio implementato dal router LVS produce dei vantaggi nei confronti dell'inoltro della richieste livello-applicazione, poichè il bilanciamento dei carichi al livello del pacchetto di rete causa un overhead minimo e permette una maggiore scalabilità.

Using scheduling, the active router can take into account the real servers' activity and, optionally, an administrator-assigned *weight* factor when routing service requests. Using assigned weights gives arbitrary priorities to individual machines. Using this form of scheduling, it is possible to create a group of real servers using a variety of hardware and software combinations and the active router can evenly load each real server.

Il meccanismo di schedulazione per LVS è fornito da una raccolta di patch del kernel chiamata module *IP Virtual Server* o *IPVS*. I suddetti moduli abilitano il cambiamento del livello di trasporto (transport layer) *Livello 4* (*L4*), ideato per funzionare bene con server multipli su di un indirizzo IP singolo.

Per controllare e direzionare i pacchetti ai real server in modo efficiente, IPVS crea una *tabella IPVS* nel kernel. Questa tabella viene utilizzata dal router LVS attivo per instradare le richieste da un indirizzo del server virtuale per, e in ritorno, dai real server nel gruppo. La tabella IPVS viene costantemente aggiornata da una utilità chiamata *ipvsadm* — aggiungendo e rimuovendo i membri del cluster a seconda della loro disponiblità.

1.3.1. Algoritmi di schedulazione

The structure that the IPVS table takes depends on the scheduling algorithm that the administrator chooses for any given virtual server. To allow for maximum flexibility in the types of services you can cluster and how these services are scheduled, Red Hat Enterprise Linux provides the following scheduling algorithms listed below. For instructions on how to assign scheduling algorithms refer to Sezione 4.6.1, «La sottosezione **SERVER VIRTUALE**».

Round-Robin Scheduling

Distribuisce ogni richiesta in modo sequenziale attraverso il gruppo di real server. Utilizzando questo algoritmo, tutti i real server vengono trattati nello stesso modo, senza considerare la capacità o il carico. Questo modello di schedulazione richiama il round-robin DNS, ma risulta essere più granulare poichè essa si basa sul collegamento alla rete e non sull'host. La schedulazione LVS round-robin non soffre lo squilibrio creato dalle richieste DNS della cache.

Weighted Round-Robin Scheduling

Distributes each request sequentially around the pool of real servers but gives more jobs to servers with greater capacity. Capacity is indicated by a user-assigned weight factor, which is then adjusted upward or downward by dynamic load information. Refer to Sezione 1.3.2, «Peso del server e Schedulazione» for more on weighting real servers.

La schedulazione Weighted round-robin è la scelta preferita se sono presenti differenze significative di capacità nei server di un gruppo. Tuttavia, se il carico richiesto varia in modo drammatico, il server con più peso potrebbe assumere un carico maggiore alle sue capacità.

Least-Connection

Distribuisce un numero maggiore di richieste ai real server con un numero minore di collegamenti attivi. Poichè mantiene un controllo sui collegamenti attivi per i real server attraverso la tabella IPVS, least-connection è un tipo di algoritmo di schedulazione dinamico ideale se è presente una

variazione molto ampia del carico. È la scelta migliore per un gruppo di real server dove ogni nodo del membro possiede più o meno la stessa capacità. Se un gruppo di server presenta una capacità diversa, allora la schedulazione weighted least-connection è quella migliore.

Weighted Least-Connections (default)

Distributes more requests to servers with fewer active connections relative to their capacities. Capacity is indicated by a user-assigned weight, which is then adjusted upward or downward by dynamic load information. The addition of weighting makes this algorithm ideal when the real server pool contains hardware of varying capacity. Refer to Sezione 1.3.2, «Peso del server e Schedulazione» for more on weighting real servers.

Locality-Based Least-Connection Scheduling

Distribuisce un numero maggiore di richieste ai server con meno collegamenti attivi in base ai loro IP di destinazione. Questo algoritmo è stato ideato per un utilizzo all'interno di un proxy-cache server cluster. Esso instrada i pacchetti di un indirizzo IP al server per quel indirizzo, a meno che il server in questione non abbia superato le proprie capacità, ed un altro server è presente con un carico a metà delle proprie capacità. In tal caso verrà assegnato l'indirizzo IP al real server con un carico minore.

Locality-Based Least-Connection Scheduling with Replication Scheduling

Distribuisce un numero maggiore di richieste ai server con un numero di collegamenti attivi minori in base ai propri IP di destinazione. Questo algoritmo è stato creato anche per un suo utilizzo in un proxy-cache server cluster. Differisce dalla schedulazione Least-Connection in base al locale, a causa di una mappatura dell'indirizzo IP target su di un sottoinsieme di nodi del real server. Le richieste vengono instradate al server all'interno del sottoinsieme, con il numero più basso di collegamenti. Se tutti i nodi per l'IP di destinazione sono oltre la loro capacità, esso replicherà un nuovo server per l'indirizzo IP di destinazione, aggiungendo il real server con un numero minore di collegamenti dal gruppo di real server al sottoinsieme di real server per l'IP di destinazione interessato. Il nodo con un carico maggiore viene rilasciato dal sottoinsieme di real server, in modo da evitare una replicazione accessiva.

Destination Hash Scheduling

Distribuisce le richieste al gruppo di real server andando alla ricerca dell'IP di destinazione in una tabella Hash statica. Questo algoritmo è stato creato per un suo utilizzo in un proxy-cache server cluster.

Source Hash Scheduling

Distribuisce le richieste al gruppo di real server andando alla ricerca dell'IP sorgente in una tabella Hash statica. Questo algoritmo è stato creato per i router LVS con firewall multipli.

1.3.2. Peso del server e Schedulazione

L'amministratore LVS è in grado di assegnare un *peso* ad ogni nodo presente nel gruppo di real server. Il suddetto peso è composto da un valore intero presente in qualsiasi algoritmo di schedulzione *weight-aware* (come ad esempio weighted least-connections), e viene utilizzato dal router LVS per indirizzare il carico in modo più uniforme su hardware con diverse capacità.

I pesi sono in relazione tra loro. Per esempio, se un real server possiede un peso uguale a 1, e l'altro server possiede un peso pari a 5, il server con peso 5 otterrà 5 collegamenti per ogni 1 collegamento ricevuto dall'altro server. Il valore di default per un real server è 1.

Anche se l'aggiunta di peso alle diverse configurazioni hardware in un gruppo di real server, può

aiutare ad un bilanciamento del carico più efficiente, questa operazione può causare squilibri temporanei quando un real server viene introdotto in un gruppo, e lo stesso è stato schedulato per usare un weighted least-connection. Per esempio, supponiamo ci siano tre server in un gruppo di real server. Server A e B hanno un peso pari a 1 ed il terzo, server C, ha un peso pari a 2. Se il server C viene per qualche ragione interrotto, i server A e B distribuiranno uniformemente il carico appena abbandonato. Tuttavia, quando il server C è nuovamente attivo, il router LVS lo considera come se avesse zero collegamenti e quindi inizia il trasferimento al server di un numero elevato di richieste in entrata fino a raggiungere una parità con i server A e B.

Per prevenire questo fenomeno gli amministratori possono trasformare il server virtuale in un server *quiesce* — ogni qualvolta un nuovo nodo del real server risulta essere online, la tabella delle least-connection viene azzerata e il router LVS instrada le richieste come se tutti i real server fossero stati appena aggiunti al cluester.

1.4. METODI D'INSTRADAMENTO

Red Hat Enterprise Linux utilizza il *Network Address Translation* o *NAT routing* per LVS, il quale conferisce all'amministratore un elevatissimo grado di flessibilità durante l'utilizzo dell'hardware disponibile, e permette l'integrazione di LVS in una rete esistente.

1.4.1. NAT Routing

Figura 1.3, «LVS Implemented with NAT Routing» , illustrates LVS utilizing NAT routing to move requests between the Internet and a private network.

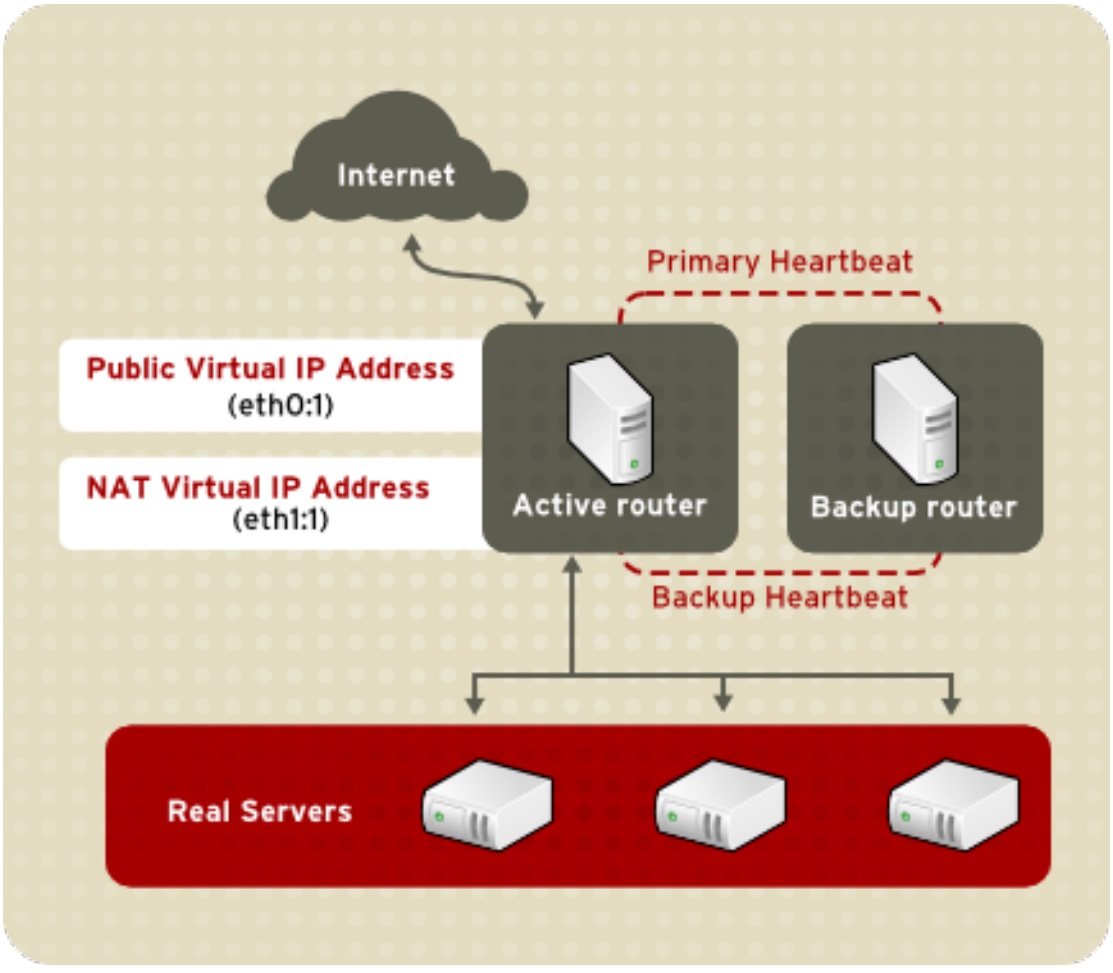


Figura 1.3. LVS Implemented with NAT Routing

Nell'esempio sono presenti due NIC nel router LVS attivo. Il NIC per internet presenta un *indirizzo IP*

reale su eth0 ed un alias dell'indirizzo IP floating su eth0:1. Il NIC per l'interfaccia di rete privata presenta un indirizzo IP reale su eth1 ed un alias dell'indirizzo IP floating su eth1:1. Se si verifica un failover, l'interfaccia virtuale a contatto con internet e quella privata rivolta verso l'interfaccia virtuale, vengono controllate simultaneamente dal router LVS di backup. Tutti i real server posizionati sulla rete privata, utilizzano l'IP floating per il router NAT come propria rotta predefinita per comunicare con il router LVS attivo, così da non compromettere la loro abilità a rispondere alle richieste provenienti da internet.

In this example, the LVS router's public LVS floating IP address and private NAT floating IP address are aliased to two physical NICs. While it is possible to associate each floating IP address to its own physical device on the LVS router nodes, having more than two NICs is not a requirement.

Utilizzando questa topografia il router LVS attivo riceve la richiesta e la direziona verso il server appropriato. Successivamente, il real server la processa e ritorna i pacchetti al router LVS, il quale utilizza il network address translation per sostituire l'indirizzo del real server nei pacchetti, con l'indirizzo VIP pubblico dei router LVS. Questo processo è chiamato *IP masquerading* poichè gli indirizzi IP dei real server sono nascosti ai client richiedenti.

Se si utilizza il NAT routing i real server potranno essere qualsiasi macchina in grado di eseguire diversi sistemi operativi. Lo svantaggio principale è rappresentato dal fatto che il router LVS potrebbe diventare il punto più debole, se utilizzato in implementazioni cluster più grandi poichè sarà necessario processare sia richieste in uscita che richieste in entrata.

1.4.2. Instradamento diretto

La creazione di una impostazione LVS che utilizzi un instradamento diretto, fornisce maggiori benefici sulle prestazioni rispetto ad altre topografie di networking LVS. L'instradamento diretto permette ai real server di processare e direzionare i pacchetti direttamente ad un utente richiedente, invece di passare tutti i pacchetti in uscita attraverso il router LVS. L'instradamento diretto riduce le problematiche relative alla prestazione di rete, relegando le funzioni del router LVS alla sola processazione dei pacchetti in ingresso.

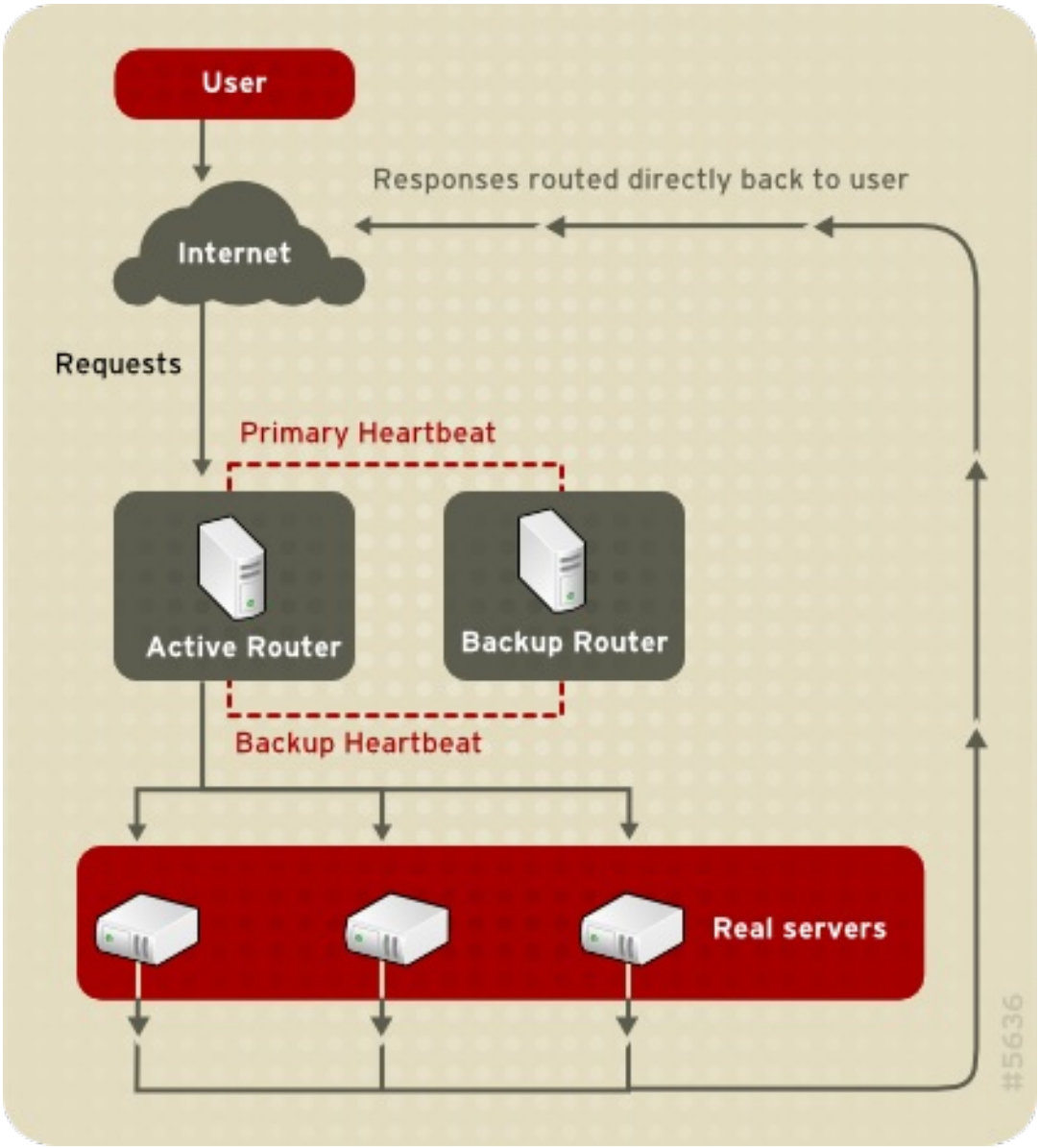


Figura 1.4. LVS Implemented with Direct Routing

In una impostazione LVS con instradamento diretto tipica, il router LVS riceve le richieste del server in arrivo attraverso il virtual IP (VIP), ed utilizza un algoritmo di schedulazione per instradare ed inviare le risposte direttamente al client bypassando i router LVS. Questo metodo d'instradamento permette di avere una certa scalabilità, poichè i real server possono essere aggiunti, senza la necessità di avere un router LVS apposito per instradare i pacchetti in uscita dal real server al client, eliminando un potenziale problema in presenza di carichi di rete molto pesanti.

1.4.2.1. Instradamento diretto e limitazioni ARP

Insieme ai numerosi vantaggi offerti dall'utilizzo dell'instradamento diretto in LVS sono presenti anche alcune limitazioni. Il problema più comune con LVS e l'instradamento diretto si verifica con l'*Address Resolution Protocol* (ARP).

In typical situations, a client on the Internet sends a request to an IP address. Network routers typically send requests to their destination by relating IP addresses to a machine's MAC address with ARP. ARP requests are broadcast to all connected machines on a network, and the machine with the correct IP/MAC address combination receives the packet. The IP/MAC associations are stored in an ARP cache, which is cleared periodically (usually every 15 minutes) and refilled with IP/MAC associations.

Il problema presente con le richieste ARP in una impostazione LVS con istradamento diretto, è

causato da una richiesta del client ad un indirizzo IP la quale deve essere associata con un indirizzo MAC per poter essere gestita. Da notare che l'indirizzo IP virtuale del sistema LVS deve anch'esso essere associato ad un MAC. Tuttavia, poichè il router LVS ed i real server hanno lo stesso VIP, la richiesta ARP verrà trasmessa a tutte le macchine associate con il VIP. Ciò potrebbe causare diversi problemi come ad esempio l'associazione del VIP direttamente ad uno dei real server, e quindi la processazione diretta delle richieste, bypassando completamente il router LVS e annullando lo scopo di una impostazione LVS.

Per risolvere questo problema fate in modo che le richieste in ingresso siano sempre inviate al router LVS, invece di essere inviate ad uno dei real server. Per fare questo utilizzate il tool di filtraggio del pacchetto **`iptables`** o **`arptables_jf`** nelle seguenti situazioni:

- **`arptables_jf`** impedisce l'associazione dei VIP con i real server da parte di ARP.
- Il metodo **`iptables`** evita completamente la problematica ARP, non configurando i VIP sui real server.

For more information on using **`arptables`** or **`iptables`** in a direct routing LVS environment, refer to Sezione 3.2.1, «Instradamento diretto e **`arptables_jf`**» or Sezione 3.2.2, «Instradamento diretto e **`iptables`**».

1.5. PERSISTENZA E FIREWALL MARK

In determinate situazioni potrebbe essere conveniente per un client ricollegarsi ripetutamente allo stesso real server, invece di avere un algoritmo di bilanciamento del carico LVS in grado di invare la richiesta al miglior server disponibile. Esempi di queste situazioni includono forme web multi-screen, cookies, SSL, e collegamenti FTP. In questi casi un client potrebbe non operare correttamente a meno che le transazioni non siano gestite dallo stesso server per garantire una certa continuità. A tale scopo LVS fornisce due diverse funzioni: la *persistenza* ed i *firewall mark*.

1.5.1. Persistenza

Quando abilitata, la persistenza funziona come un timer. E cioè quando un client si collega ad un servizio, LVS si ricorda dell'ultimo collegamento per un periodo di tempo specificato. Se lo stesso indirizzo IP del client si collega entro il suddetto periodo, esso verrà inviato allo stesso server con il quale si è precedentemente collegato — bypassando così i meccanismi di bilanciamento del carico. Se invece il suddetto collegamento si verifica dopo il periodo specificato, esso verrà gestito in base alle regole implementate.

La persistenza permette anche all'amministratore di specificare una maschera di sottorete da applicare al test dell'indirizzo IP del client, come tool per il controllo degli indirizzi con un livello elevato di persistenza, quindi raggruppando i collegamenti a quella sottorete.

Il raggruppamento dei collegamenti destinati a porte diverse può essere un processo importante per i protocolli che utilizzano più di una porta per comunicare, come ad esempio FTP. Tuttavia, la persistenza non rappresenta il modo migliore per affrontare i problemi del raggruppamento dei collegamenti destinati a porte diverse. Per queste situazioni è consigliato usare i *firewall mark*.

1.5.2. Firewall Mark

I firewall mark rappresentano un modo semplice ed efficiente per raggruppare le porte usate per un protocollo, o gruppo di protocolli, in relazione tra loro. Per esempio, se LVS viene impiegato per eseguire un sito e-commerce, i firewall mark possono essere utilizzati per raggruppare i collegamenti HTTP sulla porta 80, e rendere i collegamenti HTTP sicuri sulla porta 443. Assegnando lo stesso

firewall mark al server virtuale per ogni protocollo, è possibile conservare le informazioni sullo stato della transazione, poichè il router LVS inoltra tutte le richieste allo stesso real server dopo aver aperto un collegamento.

Grazie alla sua efficienza e facilità d'uso, gli amministratori di LVS dovrebbero usare, quando possibile, i firewall mark al posto della persistenza per raggruppare i collegamenti. Tuttavia, gli amministratori dovrebbero aggiungere la persistenza ai server virtuali insieme ai firewall mark, per far si che i client vengano ricollegati allo stesso server per un periodi di tempo adeguato.

1.6. LVS — UN DIAGRAMMA A BLOCCHI

LVS routers use a collection of programs to monitor cluster members and cluster services. Figura 1.5, «LVS Components» illustrates how these various programs on both the active and backup LVS routers work together to manage the cluster.

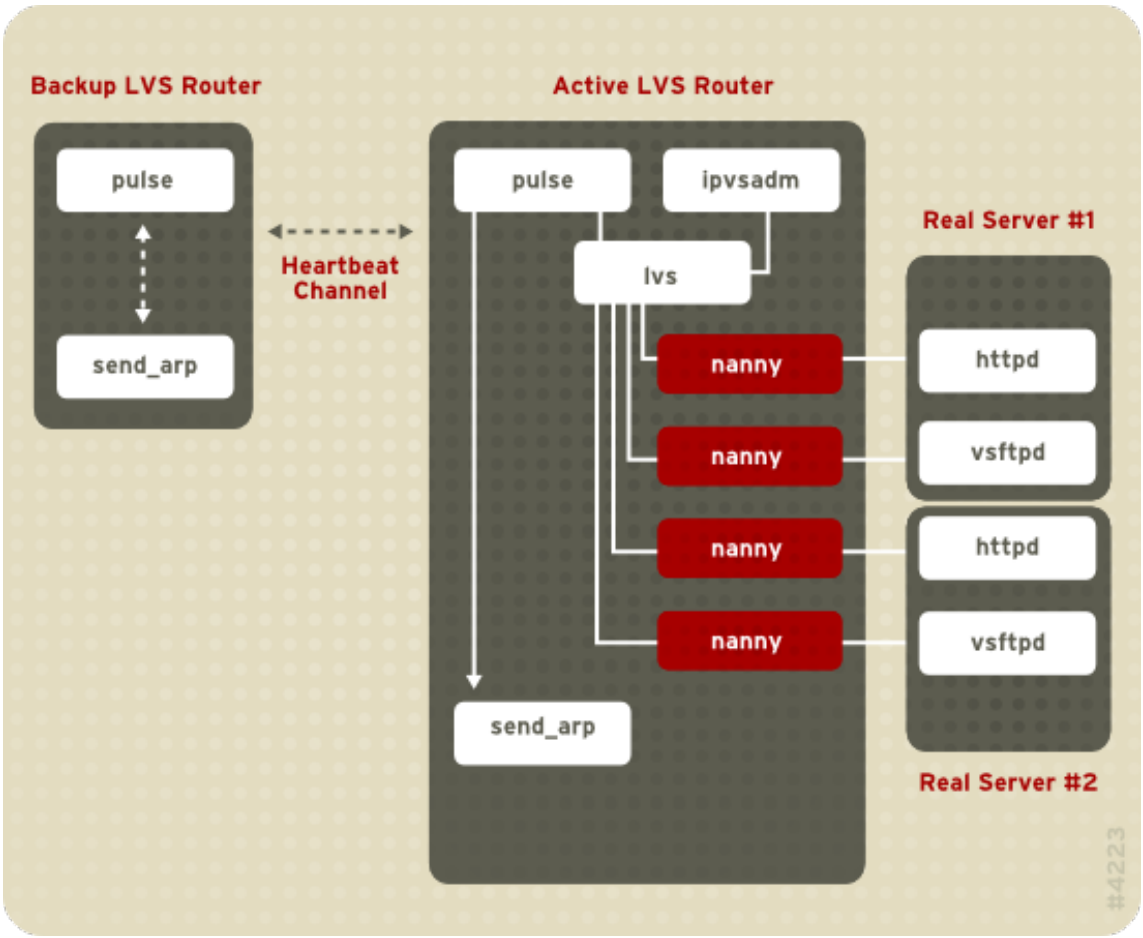


Figura 1.5. LVS Components

Il demone **pulse** viene eseguito sul router LVS passivo ed attivo. Sul router di backup **pulse** invia un *heartbeat* all'interfaccia pubblica del router attivo, per assicurarsi che il router attivo funzioni correttamente. Sul router attivo **pulse** avvia il demone **lvs**, e risponde alle interrogazioni *heartbeat* dal router LVS di backup.

Una volta avviato, il demone **lvs** richiama l'utilità **ipvsadm** per configurare e mantenere la tabella d'instradamento IPVS nel kernel, ed avvia un processo **nanny** per ogni server virtuale configurato su ogni real server. Ogni processo **nanny** controlla lo stato di un servizio configurato su di un real server, ed indica al demone **lvs** se il servizio sul real server in questione non funziona correttamente. Se viene rilevato tale malfunzionamento, il demone **lvs** indica a **ipvsadm**, di rimuovere il real server dalla tabella d'instradamento IPVS.

Se il router di backup non riceve alcuna risposta dal router attivo, verrà iniziato un processo di failover tramite **send_arp** il quale riassegna tutti gli indirizzi IP virtuali agli indirizzi hardware NIC (indirizzo *MAC*) del nodo di backup, invia un comando al router attivo tramite l'interfaccia di rete privata e pubblica per arrestare il demone **lvs** sul router attivo, ed avvia il demone **lvs** sul nodo di backup per accettare le richieste per i server virtuali configurati.

1.6.1. LVS Components

Sezione 1.6.1.1, «pulse» shows a detailed list of each software component in an LVS router.

1.6.1.1. pulse

This is the controlling process which starts all other daemons related to LVS routers. At boot time, the daemon is started by the **/etc/rc.d/init.d/pulse** script. It then reads the configuration file **/etc/sysconfig/ha/lvs.cf**. On the active router, **pulse** starts the LVS daemon. On the backup router, **pulse** determines the health of the active router by executing a simple heartbeat at a user-configurable interval. If the active router fails to respond after a user-configurable interval, it initiates failover. During failover, **pulse** on the backup router instructs the **pulse** daemon on the active router to shut down all LVS services, starts the **send_arp** program to reassign the floating IP addresses to the backup router's MAC address, and starts the **lvs** daemon.

1.6.1.2. lvs

Una volta chiamato da **pulse** il demone **lvs** viene eseguito sul router LVS attivo, legge il file di configurazione **/etc/sysconfig/ha/lvs.cf**, chiama l'utilità **ipvsadm** per compilare e mantenere la tabella d'instradamento IPVS, e assegna un processo **nanny** per ogni servizio LVS configurato. Se **nanny** riporta che un real server è inattivo, **lvs** indica alla utilità **ipvsadm**, di rimuovere il real server dalla tabella d'instradamento IPVS.

1.6.1.3. ipvsadm

Questo servizio aggiorna la tabella d'instradamento IPVS nel kernel. Il demone **lvs** imposta e gestisce LVS richiamando **ipvsadm** per aggiungere, modificare e cancellare le voci all'interno della tabella d'instradamento IPVS.

1.6.1.4. nanny

Il demone di monitoraggio **nanny** viene eseguito su ogni router LVS attivo. Attrverso questo demone il router attivo determina lo stato di ogni real server e, facoltativamente, monitorizza il carico di lavoro. Viene eseguito un processo separato per ogni servizio definito su ogni real server.

1.6.1.5. /etc/sysconfig/ha/lvs.cf

Questo è il file di configurazione di LVS. Direttamente o indirettamente tutti i demoni ottengono le proprie informazioni sulla configurazione da questo file.

1.6.1.6. Piranha Configuration Tool

Questo è il tool basato sul web per il monitoraggio, configurazione e gestione di LVS. Esso è il tool di default per il mantenimento del file di configurazione LVS **/etc/sysconfig/ha/lvs.cf**.

1.6.1.7. send_arp

Questo programma invia le trasmissioni ARP quando l'indirizzo IP floating cambia da un nodo ad un altro durante il failover.

Capitolo 2, *Configurazione LVS iniziale* reviews important post-installation configuration steps you should take before configuring Red Hat Enterprise Linux to be an LVS router.

CAPITOLO 2. CONFIGURAZIONE LVS INIZIALE

Dopo aver installato Red Hat Enterprise Linux, è necessario seguire alcuni processi di base per poter impostare i router LVS ed i real server. Questo capitolo affronta in dettaglio le suddette fasi.

NOTA

Il nodo del router LVS che diventa attivo una volta avviato il cluster, viene chiamato anche *nodo primario*. Quando configurate LVS, utilizzate il **Piranha Configuration Tool** sul nodo primario.

2.1. CONFIGURAZIONE DEI SERVIZI SUI ROUTER LVS

Il programma d'installazione di Red Hat Enterprise Linux installa tutti i componenti necessari per l'impostazione di LVS, ma i servizi appropriati dovranno essere attivati prima di eseguire la sua configurazione. Per entrambi i router LVS, impostate i servizi appropriati in modo da avviarli al momento del boot. A tal proposito sono disponibili con Red Hat Enterprise Linux tre tool primari per l'impostazione dei servizi, e quindi per la loro attivazione al momento dell'avvio: il programma della linea di comando **chkconfig**, il programma basato su ncurses **ntsysv**, ed il **Services Configuration Tool** grafico. I suddetti tool richiedono un accesso root.

NOTA

Per ottenere un accesso root aprite un prompt della shell ed utilizzate il comando **su -** seguito dalla password di root. Per esempio:

```
$ su - root password
```

Sui router LVS sono presenti tre servizi i quali dovranno essere impostati per una loro attivazione al momento dell'avvio:

- Il servizio **piranha-gui** (solo sul nodo primario)
- Il servizio **pulse**
- Il servizio **sshd**

Se state eseguendo il clustering dei servizi multi-port o se state utilizzando i firewall mark, sarà necessario abilitare anche il servizio **iptables**.

Vi consigliamo d'impostare questi servizi in modo da attivarli sui runlevel 3 e 5. Per eseguire quanto sopra descritto utilizzando il comando **chkconfig**, digitate quanto segue per ogni servizio:

/sbin/chkconfig --level 35 *daemon* on

Nel comando descritto, sostituite *daemon* con il nome del servizio che state attivando. Per ottenere un elenco dei servizi sul sistema ed il runlevel sul quale verranno attivati, emettete il seguente comando:

/sbin/chkconfig --list

AVVERTIMENTO

Turning any of the above services on using **chkconfig** does not actually start the daemon. To do this use the **/sbin/service** command. See Sezione 2.3, «Avvio del servizio del **Piranha Configuration Tool**» for an example of how to use the **/sbin/service** command.

For more information on runlevels and configuring services with **ntsysv** and the **Services Configuration Tool**, refer to the chapter titled *"Controlling Access to Services"* in the *Red Hat Enterprise Linux System Administration Guide*.

2.2. IMPOSTAZIONE PASSWORD PER IL PIRANHA CONFIGURATION TOOL

Prima di utilizzare per la prima volta il **Piranha Configuration Tool** sul router LVS primario, è necessario limitare il suo accesso attraverso la creazione di una password. Per fare questo eseguite il login come utente root ed emettete il seguente comando:

```
/usr/sbin/piranha-passwd
```

Dopo aver inserito il comando create una password amministrativa quando richiesto.

AVVERTIMENTO

Per rendere una password più sicura essa non dovrebbe contenere nomi comuni, acronimi usati frequentemente o parole presenti in qualsiasi dizionario. È vivamente sconsigliato lasciare la password non cifrata sul sistema.

Se la password viene modificata durante una sessione attiva del **Piranha Configuration Tool**, verrà richiesto all'amministratore di fornire la nuova password.

2.3. AVVIO DEL SERVIZIO DELPIRANHA CONFIGURATION TOOL

Dopo aver impostato la password per il **Piranha Configuration Tool**, avviate o riavviate il servizio **piranha-gui** situato in **/etc/rc.d/init.d/piranha-gui**. Per fare questo digitate il seguente comando come utente root:

```
/sbin/service piranha-gui start
```

or

```
/sbin/service piranha-gui restart
```

Issuing this command starts a private session of the Apache HTTP Server by calling the symbolic link

/usr/sbin/piranha_gui -> /usr/sbin/httpd. For security reasons, the **piranha-gui** version of **httpd** runs as the piranha user in a separate process. The fact that **piranha-gui** leverages the **httpd** service means that:

1. Apache HTTP Server deve essere installato sul sistema.
2. L'arresto o il riavvio di Apache HTTP Server tramite il comando **service**, arresterà il servizio **piranha-gui**.

AVVERTIMENTO

Se il comando **/sbin/service httpd stop** o **/sbin/service httpd restart** viene emesso su di un router LVS, sarà necessario avviare il servizio **piranha-gui** tramite il seguente comando:

```
/sbin/service piranha-gui start
```

The **piranha-gui** service is all that is necessary to begin configuring LVS. However, if you are configuring LVS remotely, the **sshd** service is also required. You do *not* need to start the **pulse** service until configuration using the **Piranha Configuration Tool** is complete. See Sezione 4.8, «Avvio di LVS» for information on starting the **pulse** service.

2.3.1. Come configurare la Porta del Web Server del Piranha Configuration Tool

Il **Piranha Configuration Tool** viene eseguito per default sulla porta 3636. Per modificare il numero di questa porta, cambiate la riga **Listen 3636** nella Sezione 2 del file di configurazione del **piranha-gui** web server, **/etc/sysconfig/ha/conf/httpd.conf**.

Per poter utilizzare il **Piranha Configuration Tool** avrete bisogno di almeno un web browser di solo testo. Se avviate un web browser sul router LVS primario, aprite **http://*localhost*:3636**. È possibile raggiungere il **Piranha Configuration Tool** da qualsiasi posizione tramite il web browser, semplicemente sostituendo *localhost* con l'hostname o l'indirizzo IP del router LVS primario.

Quando il vostro browser si collega al **Piranha Configuration Tool** sarà necessario registrarsi per poter accedere ai servizi di configurazione. Inserite **piranha** nel campo **Nome utente** insieme al set di password con **piranha-passwd** nel campo **Password**.

Ora che il **Piranha Configuration Tool** è in esecuzione potreste limitare l'accesso al tool attraverso la rete. La sezione seguente affronta i diversi modi attraverso i quali è possibile eseguire questo compito.

2.4. COME LIMITARE L'ACCESSO AL PIRANHA CONFIGURATION TOOL

Il **Piranha Configuration Tool** necessita di una combinazione valida di nome utente e password. Tuttavia, tutti i dati passati al **Piranha Configuration Tool** sono in testo semplice, e per questo otivo è consigliato limitare l'accesso solo alle reti fidate o alla macchina locale.

The easiest way to restrict access is to use the Apache HTTP Server's built in access control mechanisms by editing **/etc/sysconfig/ha/web/secure/.htaccess**. After altering the file you do not have to restart the **piranha-gui** service because the server checks the **.htaccess** file each

time it accesses the directory.

Per default i controlli per l'accesso a questa directory permettono a qualsiasi utente di visualizzare i contenuti presenti. Di seguito viene riportato l'accesso predefinito:

```
Order deny,allow
Allow from all
```

Per limitare l'accesso del **Piranha Configuration Tool** al solo localhost, modificate il file **`.htaccess`** in modo da permettere l'accesso solo dal dispositivo loopback (127.0.0.1). Per maggiori informazioni sul dispositivo loopback consultate il capitolo *Script di rete*nella *Red Hat Enterprise Linux Reference Guide*

```
Order deny,allow
Deny from all
Allow from 127.0.0.1
```

Come riportato in questo esempio è possibile abilitare anche host specifici o sottoreti:

```
Order deny,allow
Deny from all
Allow from 192.168.1.100
Allow from 172.16.57
```

In questo esempio solo i web browser della macchina con un indirizzo IP 192.168.1.100, e macchine presenti sulla rete 172.16.57/2 saranno in grado di accedere al **Piranha Configuration Tool**.

AVVERTIMENTO

La modifica del file **`.htaccess`** del **Piranha Configuration Tool** limiterà l'accesso alle pagine di configurazione nella directory **`/etc/sysconfig/ha/web/secure/`**, ma non alle pagine di login e d'aiuto in **`/etc/sysconfig/ha/web/`**. Per limitare l'accesso alla suddetta directory, create un file **`.htaccess`** nella directory **`/etc/sysconfig/ha/web/`** con le righe **`order`**, **`allow`**, e **`deny`** in modo identico a **`/etc/sysconfig/ha/web/secure/.htaccess`**.

2.5. ABILITAZIONE DEL PACKET FORWARDING

Per permettere al router LVS di inoltrare i pacchetti di rete in modo corretto ai real server, è necessario aver abilitato sul nodo del router LVS nel kernel l'IP forwarding. Registratevi come utente root e modificate la riga **`net.ipv4.ip_forward = 0`** in **`/etc/sysctl.conf`** nel modo seguente;

```
net.ipv4.ip_forward = 1
```

Le modifiche verranno implementate durante il successivo riavvio del sistema.

Per controllare se l'IP forwarding è stato abilitato, emettete il seguente comando come utente root:

`/sbin/sysctl net.ipv4.ip_forward`

Se il comando ritorna **1** l'IP forwarding è stato abilitato. Se invece il valore è **0**, allora sarà possibile abilitarlo usando il seguente comando:

```
/sbin/sysctl -w net.ipv4.ip_forward=1
```

2.6. CONFIGURAZIONE DEI SERVIZI SUI REAL SERVER

Se i real server risultano essere sistemi Red Hat Enterprise Linux, impostate i demoni del server appropriati in modo da attivarsi al momento dell'avvio. I suddetti demoni possono includere **httpd** per servizi web o **xinetd** per servizi Telnet o FTP.

Potrebbe essere necessario accedere i real server in modo remoto, e per questo motivo è consigliato installare ed eseguire il demone **sshd**.

CAPITOLO 3. IMPOSTAZIONE DEL LVS

LVS consiste in due gruppi di base: i router LVS ed i real server. Per evitare il verificarsi di un single point of failure, ogni gruppo dovrebbe avere almeno due sistemi membri.

Il gruppo del router LVS dovrebbe consistere in due sistemi identici o molto simili sui quali viene eseguito Red Hat Enterprise Linux. Uno che funge come router LVS attivo e l'altro in grado di essere in modalità di hot standby, quindi entrambi dovranno avere una capacità quanto più simile possibile.

Prima di selezionare e configurare l'hardware per il gruppo del real server, è necessario determinare quale topologia LVS usare.

3.1. LA RETE NAT LVS

La topologia NAT vi permette di utilizzare l'hardware esistente in diversi modi, ma risulta essere limitato nella propria abilità di gestire carichi molto grandi, a causa del passaggio di tutti i pacchetti in entrata e in uscita attraverso il router LVS.

Struttura della rete

Da una prospettiva della struttura di rete, la topologia per LVS che utilizza un NAT routing risulta essere quella più semplice da configurare, poichè è necessario solo un punto d'accesso alla rete pubblica. I real server ritornano tutte le richieste attraverso il router LVS in modo da essere nella propria rete privata.

Hardware

La topologia NAT è la più flessibile in relazione all'hardware, poichè i real server non devono essere macchine Linux per poter funzionare correttamente. In una topologia NAT ogni real server avrà bisogno di un solo NIC poichè esso risponderà solo al router LVS. I router LVS, invece, avranno bisogno ciascuno di due NIC per instradare il traffico tra le due reti. Poichè questa topologia crea una limitazione nei confronti del router LVS, è possibile impiegare i NIC gigabit Ethernet su ogni router LVS per migliorare la larghezza di banda che un router LVS è in grado di gestire. Se il gigabit Ethernet viene implementato sui router LVS, qualsiasi interruttore che collega i real server ai router LVS, deve avere almeno due porte gigabit Ethernet per gestire in modo efficiente il carico.

Software

Poichè la topologia NAT richiede l'utilizzo di **`iptables`** per alcune configurazioni, potrebbe esser necessario eseguire una ulteriore configurazione esterna al **Piranha Configuration Tool**. In particolare, i servizi FTP e l'utilizzo dei firewall mark richiedono una configurazione manuale supplementare dei router LVS, per instradare in modo correto tutte le richieste.

3.1.1. Configurazione delle interfacce di rete per LVS con NAT

To set up LVS with NAT, you must first configure the network interfaces for the public network and the private network on the LVS routers. In this example, the LVS routers' public interfaces (**`eth0`**) will be on the 192.168.26/24 network (I know, I know, this is not a routable IP, but let us pretend there is a firewall in front of the LVS router for good measure) and the private interfaces which link to the real servers (**`eth1`**) will be on the 10.11.12/24 network.

So on the active or *primary* LVS router node, the public interface's network script, **`/etc/sysconfig/network-scripts/ifcfg-eth0`**, could look something like this:

```
DEVICE=eth0
```

```
BOOTPROTO=static
ONBOOT=yes
IPADDR=192.168.26.9
NETMASK=255.255.255.0
GATEWAY=192.168.26.254
```

/etc/sysconfig/network-scripts/ifcfg-eth1 per l'interfaccia NAT privata sul router LVS potrebbe somigliare al seguente:

```
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
IPADDR=10.11.12.9
NETMASK=255.255.255.0
```

In this example, the VIP for the LVS router's public interface will be 192.168.26.10 and the VIP for the NAT or private interface will be 10.11.12.10. So, it is essential that the real servers route requests back to the VIP for the NAT interface.

IMPORTANTE

The sample Ethernet interface configuration settings in this section are for the real IP addresses of an LVS router and *not* the floating IP addresses. To configure the public and private floating IP addresses the administrator should use the **Piranha Configuration Tool**, as shown in Sezione 4.4, «**GLOBAL SETTINGS**» and Sezione 4.6.1, «La sottosezione **SERVER VIRTUALE**».

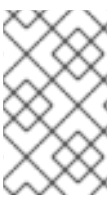
After configuring the primary LVS router node's network interfaces, configure the backup LVS router's real network interfaces — taking care that none of the IP address conflict with any other IP addresses on the network.

IMPORTANTE

Assicuratevi che ogni interfaccia sul nodo di backup serva la stessa rete dell'interfaccia presente sul nodo primario. Per esempio se eth0 si collega alla rete pubblica sul nodo primario, la stessa deve collegarsi anche alla rete pubblica sul nodo di backup.

3.1.2. Instradamento sui real server

La cosa più importante da ricordare durante la configurazione delle interfacce di rete dei real server in una topologia NAT, è quella d'impostare il gateway per l'indirizzo IP floating NAT del router LVS. In questo esempio, l'indirizzo in questione sarà 10.11.12.10.



NOTA

Once the network interfaces are up on the real servers, the machines will be unable to ping or connect in other ways to the public network. This is normal. You will, however, be able to ping the real IP for the LVS router's private interface, in this case 10.11.12.8.

So the real server's **/etc/sysconfig/network-scripts/ifcfg-eth0** file could look similar to this:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=10.11.12.1
NETMASK=255.255.255.0
GATEWAY=10.11.12.10
```

AVVERTIMENTO

Se un real server possiede più di una interfaccia di rete configurata con la riga **GATEWAY=**, la prima interfaccia usata otterrà il gateway. Per questo motivo se **eth0** e **eth1** sono configurate, e **eth1** viene utilizzata per LVS, i real server potrebbero non indirizzare le richieste in modo corretto.

È fortemente consigliato disabilitare le interfacce di rete estranee tramite l'impostazione di **ONBOOT=no**, negli script di rete interessati all'interno della directory **/etc/sysconfig/network-scripts/**, oppure assicuratevi che il gateway sia stato impostato correttamente nella prima interfaccia usata.

3.1.3. Abilitzione del NAT Routing sui router LVS

In a simple NAT LVS configuration where each clustered service uses only one port, like HTTP on port 80, the administrator needs only to enable packet forwarding on the LVS routers for the requests to be properly routed between the outside world and the real servers. See Sezione 2.5, «Abilitazione del Packet Forwarding» for instructions on turning on packet forwarding. However, more configuration is necessary when the clustered services require more than one port to go to the same real server during a user session. For information on creating multi-port services using firewall marks, see Sezione 3.4, «Servizi multi-port e LVS» .

Once forwarding is enabled on the LVS routers and the real servers are set up and have the clustered services running, use the **Piranha Configuration Tool** to configure LVS as shown in Capitolo 4, *Configurazione dei router LVS con**Piranha Configuration Tool***.

AVVERTIMENTO

Do not configure the floating IP for **eth0:1** or **eth1:1** by manually editing network scripts or using a network configuration tool. Instead, use the **Piranha Configuration Tool** as shown in Sezione 4.4, «**GLOBAL SETTINGS**» and Sezione 4.6.1, «La sottosezione **SERVER VIRTUALE**».

When finished, start the **pulse** service as shown in Sezione 4.8, «Avvio di LVS» . Once **pulse** is up and running, the active LVS router will begin routing requests to the pool of real servers.

3.2. LVS TRAMITE UN INSTRADAMENTO DIRETTO

As mentioned in Sezione 1.4.2, «Instradamento diretto», direct routing allows real servers to process and route packets directly to a requesting user rather than passing outgoing packets through the LVS router. Direct routing requires that the real servers be physically connected to a network segment with the LVS router and be able to process and direct outgoing packets as well.

Struttura della rete

In una impostazione per il routing LVS diretto, il router LVS avrà bisogno di ricevere le richieste in entrata e direzionarle al real server corretto per una loro processazione. I real server a loro volta dovranno instradare *direttamente* le risposte al client. Ciò significa che se il client è su internet ed invia il pacchetto ad un real server attraverso il router LVS, il real server dovrà essere in grado di contattare direttamente il client attraverso internet. È possibile eseguire tale processo attraverso la configurazione del gateway per il real server, in modo da passare i pacchetti ad internet. Ogni real server nel pool dei server, può avere un proprio gateway separato (e ogni gateway può avere un proprio collegamento ad internet), permettendo una portata e scalabilità massima. Tuttavia per impostazioni LVS tipiche i real server possono comunicare attraverso un gateway (e quindi un collegamento di rete).

IMPORTANTE

Non è consigliato utilizzare il router come gateway per i real server, in quanto è possibile creare una impostazione più complessa e non necessaria, insieme ad un carico di rete sul router LVS, i quali daranno luogo a limitazioni presenti nel NAT routing.

Hardware

I requisiti hardware di un sistema LVS che utilizza un instradamento diretto sono simili ad altre topologie LVS. Mentre il router LVS necessita di eseguire Red Hat Enterprise Linux per processare le richieste in entrata ed eseguire il bilanciamento del carico per i real server, i real server non hanno bisogno di essere machine Linux per funzionare correttamente. I router LVS necessitano ciascuno di uno o due NIC (se è presente un router di backup). Potrete utilizzare due NIC per facilitare la configurazione e per separare in modo netto il traffico — le richieste in ingresso vengono gestite da un solo NIC, ed i pacchetti indirizzati ai real server gestiti dal secondo.

Poichè i real server bypassano il router LVS ed inviano i pacchetti in uscita direttamente ad un client, sarà necessario un gateway per Internet. Per una massima disponibilità e prestazione, ogni real server può essere collegato alla rete portante alla quale il client è collegato (come ad esempio Internet o intranet).

Software

There is some configuration outside of **Piranha Configuration Tool** that needs to be done, especially for administrators facing ARP issues when using LVS via direct routing. Refer to Sezione 3.2.1, «Instradamento diretto e `arptables_jf`» or Sezione 3.2.2, «Instradamento diretto e `iptables`» for more information.

3.2.1. Instradamento diretto e `arptables_jf`

In order to configure direct routing using **`arptables_jf`**, each real server must have their virtual IP address configured, so they can directly route packets. ARP requests for the VIP are ignored entirely by the real servers, and any ARP packets that might otherwise be sent containing the VIPs are mangled to contain the real server's IP instead of the VIPs.

Utilizzando il metodo **arptables_jf** le applicazioni potrebbero legarsi ad ogni VIP o porta, serviti dal real server. Per esempio, il metodo **arptables_jf** permette a istanze multiple di Apache HTTP Server di essere eseguite esplicitamente su VIP diversi del sistema. L'utilizzo di **arptables_jf** comporta vantaggi significativi in termini di prestazione rispetto all'opzione **iptables**.

Tuttavia utilizzando il metodo **arptables_jf** i VIP non potranno essere configurati per avviarsi al boot, utilizzando i tool di configurazione del sistema di Red Hat Enterprise Linux.

Per configurare ogni real server in modo da ignorare le richieste ARP per ogni indirizzo IP virtuale, seguite le fasi di seguito riportate:

1. Create le entry della tabella ARP per ogni indirizzo IP virtuale su ogni real server (il real_ip è l'IP usato dal director per comunicare con il real server; spesso è l'IP bound per **eth0**):

   ```
   arptables -A IN -d <virtual_ip> -j DROP
   arptables -A OUT -s <virtual_ip> -j mangle --mangle-ip-s <real_ip>
   ```

   Ciò permetterà ai real server di ignorare tutte le richieste ARP per gli indirizzi IP virtuali, modificando qualsiasi risposta ARP in uscita che potrebbe contenere l'IP virtuale se non modificata, contenendo invece l'IP reale del server. L'unico nodo che dovrebbe rispondere alle richieste ARP per qualsiasi VIP, risulta essere il nodo LVS attivo.

2. Una volta completato questo processo su ogni real server, salvate le entry per la tabella ARP digitando i seguenti comandi su ogni real server:

   service arptables_jf save

   chkconfig --level 2345 arptables_jf on

   Il comando **chkconfig** causerà il ricaricamento della configurazione di arptables al momento dell'avvio da parte del sistema — prima dell'avvio della rete.

3. Configurate l'indirizzo IP virtuale su tutti i real server utilizzando **ifconfig** per creare un alias IP. Per esempio:

   ```
   # ifconfig eth0:1 192.168.76.24 netmask 255.255.252.0 broadcast
   192.168.79.255 up
   ```

   Oppure utilizzando la utilità **iproute2 ip**, per esempio:

   ```
   # ip addr add 192.168.76.24 dev eth0
   ```

   Come precedentemente riportato, gli indirizzi IP virtuali non possono essere configurati in modo da aviarsi al boot utilizzando i tool di configurazione del sistema di Red Hat. Una soluzione a questo problema potrebbe essere quella di posizionare i suddetti comandi in **/etc/rc.d/rc.local**.

4. Configure Piranha for Direct Routing. Refer to Capitolo 4, *Configurazione dei router LVS con Piranha Configuration Tool* for more information.

3.2.2. Instradamento diretto e iptables

È possibile risolvere il problema riguardante ARP durante l'utilizzo del metodo d'instradamento diretto, tramite la creazione delle regole firewall di **iptables**. Per configurare l'instradamento diretto

utilizzando **iptables**, è necessario utilizzare regole in grado di creare un proxy trasparente, e che permettono quindi ad un real server di servire i pacchetti inviati all'indirizzo VIP, anche se l'indirizzo VIP non è presente sul sistema.

Il metodo **iptables** è più semplice da configurare rispetto al metodo **arptables_jf**. Questo metodo è in grado di evitare interamente le problematiche LVS ARP, poichè gli indirizzi IP virtuali esistono solo sul director LVS attivo.

Tuttavia sono presenti alcune problematiche relative alla prestazione se si utilizza il metodo **iptables** rispetto a **arptables_jf**. Tali problematiche sono dovute ad un overhead nel forwarding/masquerading di ogni pacchetto.

Non è possibile altresì riutilizzare le porte usando il metodo **iptables**. Per esempio, non è possibile eseguire due servizi Apache HTTP Server separati ma uniti alla porta 80, poichè entrambi devono essere uniti a ***INADDR_ANY*** invece degli indirizzi IP virtuali.

Per configurare l'instradamento diretto utilizzando il metodo **iptables**, seguite le fasi di seguito riportate:

1. Su ogni real server eseguite il seguente comando per ogni combinazione di protocollo (TCP o UDP), VIP, e porta da servire:

   iptables -t nat -A PREROUTING -p <tcp|udp> -d <vip> --dport <port> -j REDIRECT

   Questo comando causerà la processazione da parte dei real server, dei pacchetti destinati al VIP e alla porta a loro conferiti.

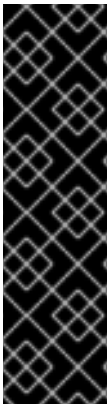
2. Salvate la configurazione su ogni real server:

   ```
   # service iptables save
   # chkconfig --level 2345 iptables on
   ```

   I comandi sopra riportati causano il ricaricamento della configurazione di **iptables** al momento dell'avvio — prima dell'avvio della rete.

3.3. COME CREARE LA CONFIGURAZIONE

Dopo aver determinato quale metodo implementare, sarà necessario collegare l'hardware sulla rete.



IMPORTANTE

I dispositivi dell'adattatore sui router LVS devono essere configurati in modo da accedere alle stesse reti. Per esempio se **eth0** si collega alla rete pubblica, e **eth1** si collega a quella privata, allora gli stessi dispositivi sul router LVS di backup dovranno collegarsi alle stesse reti.

Il gateway presente nella prima interfaccia visualizzata al momento dell'avvio viene aggiunto alla tabella d'instradamento, mentre i gateway seguenti presenti su altre interfacce vengono ignorati. Considerate tale comportamento durante la configurazione dei real server.

Dopo aver collegato fisicamente l'hardware, configurate le interfacce di rete sui router LVS di backup e primari. È possibile eseguire tale operazione usando un'applicazione grafica come ad esempio

system-config-network, oppure modificando manualmente gli script di rete. Per maggiori informazioni su come aggiungere i dispositivi utilizzando **system-config-network**, consultate il capitolo *Configurazione di rete*nella *Red Hat Enterprise Linux Deployment Guide* Per il rimando di questo capitolo, gli esempi di alterazioni delle interfacce di rete vengono eseguiti manualmente o attraverso il **Piranha Configuration Tool**.

3.3.1. Suggerimenti generali per LVS Networking

Configurate gli indirizzi IP reali per la rete pubblica e privata sui router LVS, prima di configurare LVS utilizzando il **Piranha Configuration Tool**. Le sezioni su ogni topologia presentano degli esempi di indirizzi di rete, vi preghiamo di utilizzare indirizzi di rete validi. Di seguito sono riportati alcuni comandi molto utili per l'attivazione delle interfacce di rete o per il controllo del loro stato.

Attivazione delle interfacce di rete reali

Per attivare una interfaccia di rete reale, utilizzate il seguente comando come utente root, sostituendo *N* con il numero corrispondente all'interfaccia (`eth0` e `eth1`).

```
/sbin/ifup ethN
```

AVVERTIMENTO

Do *not* use the `ifup` scripts to bring up any floating IP addresses you may configure using **Piranha Configuration Tool** (`eth0:1` or `eth1:1`). Use the `service` command to start `pulse` instead (see Sezione 4.8, «Avvio di LVS» for details).

Disattivazione delle interfacce di rete reali

Per disattivare una interfaccia di rete reale utilizzate il seguente comando come utente root, sostituendo *N* con il numero corrispondente all'interfaccia (`eth0` e `eth1`).

```
/sbin/ifdown ethN
```

Controllo dello stato delle interfacce di rete

Se desiderate controllare quale interfaccia è attiva in un determinato momento, digitate quanto segue:

```
/sbin/ifconfig
```

Per visualizzare la tabella d'instradamento per una macchina emettete il seguente comando:

```
/sbin/route
```

3.4. SERVIZI MULTI-PORT E LVS

LVS routers under any topology require extra configuration when creating multi-port LVS services. Multi-port services can be created artificially by using firewall marks to bundle together different, but related protocols, such as HTTP (port 80) and HTTPS (port 443), or when LVS is used with true multi-

port protocols, such as FTP. In either case, the LVS router uses firewall marks to recognize that packets destined for different ports, but bearing the same firewall mark, should be handled identically. Also, when combined with persistence, firewall marks ensure connections from the client machine are routed to the same host, as long as the connections occur within the length of time specified by the persistence parameter. For more on assigning persistence to a virtual server, see Sezione 4.6.1, «La sottosezione SERVER VIRTUALE».

Sfortunatamente il meccanismo usato per bilanciare il carico sui real server — IPVS — è in grado di riconoscere i firewall mark assegnati ad un pacchetto, ma non di assegnare i firewall mark. Il compito di *assegnare* i firewall mark deve essere espletato dal filtro del pacchetto di rete, **iptables**, esternamente al **Piranha Configuration Tool**.

3.4.1. Assegnazione dei firewall mark

Per assegnare i firewall mark ad un pacchetto destinato ad una porta particolare, l'amministratore dovrà utilizzare **iptables**.

This section illustrates how to bundle HTTP and HTTPS as an example; however, FTP is another commonly clustered multi-port protocol. If an LVS is used for FTP services, refer to Sezione 3.5, «Configurazione di FTP» for configuration details.

La regola di base da ricordare quando si utilizzano i firewall mark e quella che per ogni protocollo che utilizza un firewall mark con il **Piranha Configuration Tool**, è necessaria la presenza di una regola **iptables** equivalente per assegnare i firewall mark ai pacchetti di rete.

Prima di creare le regole per il filtro dei pacchetti di rete, assicuratevi che non siano presenti altre regole. Per fare questo aprite un prompt della shell, registratevi come utente root e digitate:

```
/sbin/service iptables status
```

Se **iptables** non è in esecuzione, il prompt apparirà istantaneamente.

Se **iptables** è attivo, esso visualizzerà un set di regole. Se sono presenti alcune regole, digitate il seguente comando:

```
/sbin/service iptables stop
```

Se le regole già presenti sono regole importanti, controllate il contenuto di **/etc/sysconfig/iptables** e copiate qualsiasi regola importante in un luogo sicuro prima di procedere.

Di seguito vengono riportate le regole che assegnano lo stesso firewall mark, 80, al traffico in ingresso destinato all'indirizzo IP floating, *n.n.n.n*, su porte 80 e 443.

```
/sbin/modprobe ip_tables
```

```
/sbin/iptables -t mangle -A PREROUTING -p tcp -d n.n.n.n/32 --dport 80 -j
MARK --set-mark 80
```

```
/sbin/iptables -t mangle-A PREROUTING -p tcp -d n.n.n.n/32 --dport 443 -j
MARK --set-mark 80
```

For instructions on assigning the VIP to the public network interface, see Sezione 4.6.1, «La sottosezione SERVER VIRTUALE». Also note that you must log in as root and load the module for **iptables** before issuing rules for the first time.

Nei comandi **`iptables`** sopra riportati, *n.n.n.n* deve essere sostituito con l'IP floating per i vostri server virtuali HTTP e HTTPS. Questi comandi avranno l'effetto di assegnare a qualsiasi traffico indirizzato al VIP sulle porte appropriate, un firewall mark di 80, il quale a sua volta viene riconosciuto da IPVS ed inoltrato in modo appropriato.

AVVERTIMENTO

The commands above will take effect immediately, but do not persist through a reboot of the system. To ensure network packet filter settings are restored upon reboot, refer to Sezione 3.6, «Come salvare le impostazioni per il filtro del pacchetto di rete»

3.5. CONFIGURAZIONE DI FTP

Il File Transport Protocol (FTP) è un protocollo multi-port vecchio e complesso il quale presenta un set ben distinto di prove per un ambiente LVS. Per capirne la natura di queste prove, è necessario comprendere il funzionamento di FTP.

3.5.1. Come funziona FTP

Con la maggior parte dei rapporti intrapresi con altri client server, la macchina client apre un collegamento per il server attraverso una determinata porta, la quale verrà utilizzata dal server per rispondere al client. Quando un client FTP si collega ad un server FTP, esso aprirà un collegamento con la porta 21 di controllo FTP. Subito dopo il *client* indicherà al *server* FTP se stabilire un collegamento *attivo* o *passivo*. Il tipo di collegamento scelto dal client determina il tipo di risposta del server, e la porta attraverso la quale avverrà tale transazione.

I due tipi di collegamento dati sono:

Collegamenti attivi

Quando un collegamento attivo viene stabilito il *server* aprirà un collegamento dati per il client, dalla porta 20 ad una porta con un valore più alto sulla macchina client. Successivamente tutti i dati del server vengono poi passati attraverso questo collegamento.

Collegamenti passivi

Quando stabilite un collegamento passivo, il *client* richiederà al server FTP di stabilire una porta per il collegamento passivo, la quale può essere superiore a 10,000. Per questa sezione il server a si unisce alla suddetta porta, riferendo al client il numero della porta stessa. A questo punto il client aprirà la nuova porta per il collegamento dati. Ogni richiesta fatta dal client risulterà in un collegamento dati separato. La maggior parte dei client FTP moderni cercano di stabilire un collegamento passivo al momento della richiesta dei dati dal server.

NOTA

Il *client* determina il tipo di collegamento, e non il server. Ciò significa che per eseguire in modo effettivo il cluster FTP, è necessario configurare i router LVS in modo da gestire sia i collegamenti attivi che quelli passivi.

Il rapporto client/server FTP è in grado di aprire un numero molto grande di porte, le quali sono sconosciute a IPVS ed al **Piranha Configuration Tool**.

3.5.2. Come tutto questo influenza il routing LVS

L'inoltro del pacchetto da parte di IPVS abilita solo i collegamenti in ingresso ed in uscita dal cluster in base al riconoscimento del proprio numero di porta o del firewall mark. Se un client esterno al cluster cerca di aprire una porta che IPVS non è configurato a gestire, esso interrompe il collegamento. In modo simile, se il real server cerca di aprire un collegamento su internet attraverso una porta che IPVS non conosce, esso interromperà il collegamento. Ciò significa che *tutti* i collegamenti provenienti dai client FTP su internet, *devono* avere lo stesso firewall mark, e tutti i collegamenti provenienti dal server FTP *devono* essere inoltrati correttamente ad internet utilizzando le regole di filtraggio dei pacchetti di rete.

3.5.3. Creazione delle regole per il filtro dei pacchetti di rete

Before assigning any **iptables** rules for FTP service, review the information in Sezione 3.4.1, «Assegnazione dei firewall mark» concerning multi-port services and techniques for checking the existing network packet filtering rules.

Below are rules which assign the same firewall mark, 21, to FTP traffic. For these rules to work properly, you must also use the **VIRTUAL SERVER** subsection of **Piranha Configuration Tool** to configure a virtual server for port 21 with a value of **21** in the **Firewall Mark** field. See Sezione 4.6.1, «La sottosezione SERVER VIRTUALE» for details.

3.5.3.1. Regole per collegamenti attivi

Le regole per i collegamenti attivi indicano al kernel di accettare ed inoltrare i collegamenti in arrivo nell'indirizzo IP floating *interno* sulla porta 20 — la porta dei dati FTP.

Il seguente comando **iptables** permette al router LVS di accettare i collegamenti in uscita dai real server non conosciuti da IPVS:

```
/sbin/iptables -t nat -A POSTROUTING -p tcp -s n.n.n.0/24 --sport 20 -j MASQUERADE
```

In the **iptables** command, *n.n.n* should be replaced with the first three values for the floating IP for the NAT interface's internal network interface defined in the **GLOBAL SETTINGS** panel of **Piranha Configuration Tool**.

3.5.3.2. Regole per i collegamenti passivi

Le regole per i collegamenti passivi assegnano il firewall mark appropriato ai collegamenti in entrata provenienti da internet per l'IP floating del servizio, su di una determinata gamma di porte — da 10,000 a 20,000.

AVVERTIMENTO

Se state limitando la gamma di porte per i collegamenti passivi, sarà necessario configurare il server VSFTP in modo da utilizzare una gamma di porte corrispondente. È possibile eseguire tale operazione aggiungendo le seguenti righe su **/etc/vsftpd.conf**:

```
pasv_min_port=10000

pasv_max_port=20000
```

È necessario controllare altresì l'indirizzo mostrato dal server al client per i collegamenti FTP passivi. In un sistema LVS con un NAT routing, aggiungete la seguente riga su **/etc/vsftpd.conf**, per sovrascrivere l'indirizzo IP del real server sul VIP, il quale viene visualizzato dal client al momento del collegamento. Per esempio:

```
pasv_address=n.n.n.n
```

Sostituite *n.n.n.n* con l'indirizzo VIP del sistema LVS.

Per la configurazione di altri server FTP, consultate le rispettive documentazioni.

Questa gamma dovrebbe essere abbastanza grande per la maggior parte delle situazioni; tuttavia è possibile aumentare questo numero in modo da includere tutte le porte non sicure, tramite la modifica di **10000:20000** nei comandi sotto riportati in **1024:65535**.

I seguenti comandi **iptables** avranno l'effetto di assegnare un firewall mark di 21, a qualsiasi traffico indirizzato all'IP floating delle porte appropriate, il quale a sua volta viene riconosciuto da IPVS ed inoltrato in modo appropriato:

```
/sbin/iptables -t mangle -A PREROUTING -p tcp -d n.n.n.n/32 --dport 21 -j
MARK --set-mark 21
```

```
/sbin/iptables -t mangle -A PREROUTING -p tcp -d n.n.n.n/32 --dport
10000:20000 -j MARK --set-mark 21
```

Nei comandi **iptables**, *n.n.n.n* dovrebbe esser sostituito con l'IP floating per il server virtuale FTP, definito nella sottosezione **SERVER VIRTUALE** del **Piranha Configuration Tool**.

AVVERTIMENTO

The commands above take effect immediately, but do not persist through a reboot of the system. To ensure network packet filter settings are restored after a reboot, see Sezione 3.6, «Come salvare le impostazioni per il filtro del pacchetto di rete»

Finally, you need to be sure that the appropriate service is set to activate on the proper runlevels. For more on this, refer to Sezione 2.1, «Configurazione dei servizi sui router LVS».

3.6. COME SALVARE LE IMPOSTAZIONI PER IL FILTRO DEL PACCHETTO DI RETE

Dopo aver configurato i filtri del pacchetto di rete appropriato, salvate le impostazioni in modo da ripristinarli dopo il riavvio. Per **iptables**, digitate il seguente comando:

```
/sbin/service iptables save
```

Esso salverà le impostazioni in **/etc/sysconfig/iptables**, e potranno essere richiamate al momento dell'avvio.

Once this file is written, you are able to use the **/sbin/service** command to start, stop, and check the status (using the status switch) of **iptables**. The **/sbin/service** will automatically load the appropriate module for you. For an example of how to use the **/sbin/service** command, see Sezione 2.3, «Avvio del servizio del **Piranha Configuration Tool**».

Finally, you need to be sure the appropriate service is set to activate on the proper runlevels. For more on this, see Sezione 2.1, «Configurazione dei servizi sui router LVS».

Il capitolo successivo spiega come utilizzare il **Piranha Configuration Tool** per configurare il router LVS, e descrive le fasi necessarie per un LVS attivo.

CAPITOLO 4. CONFIGURAZIONE DEI ROUTER LVS CON PIRANHA CONFIGURATION TOOL

Il **Piranha Configuration Tool** fornisce un approccio strutturato alla creazione del file di configurazione necessario per LVS — `/etc/sysconfig/ha/lvs.cf`. Questo capitolo descrive l'operazione di base del **Piranha Configuration Tool**, ed il processo di attivazione dell'LVS una volta completata la configurazione.

IMPORTANTE

Il file di configurazione per LVS segue regole molto precise di formattazione. L'utilizzo del **Piranha Configuration Tool** è il modo migliore per prevenire errori di sintassi in `lvs.cf` e quindi prevenire errori software.

4.1. SOFTWARE NECESSARIO

Per poter utilizzare il **Piranha Configuration Tool** il servizio `piranha-gui` deve essere in esecuzione sul router LVS primario. Per configurare LVS sarà necessario almeno un web browser di solo testo, come ad esempio `links`. Se desiderate accedere al router LVS da un'altra machcina, allora avrete bisogno anche di un collegamento `ssh` per il router LVS primario come utente root.

Durante la configurazione del router LVS primario è consigliato mantenere un collegamento `ssh` corrente in una finestra del terminale. Questo collegamento fornisce un modo sicuro per riavviare `pulse` ed altri servizi, configurare i filtri del pacchetto di rete, e monitorare `/var/log/messages` durante la risoluzione dei problemi.

Le quattro sezioni successive vi guideranno attraverso le pagine di configurazione del **Piranha Configuration Tool**, fornendo le istruzioni sul suo utilizzo per l'impostazione di LVS.

4.2. ACCESSO AL PIRANHA CONFIGURATION TOOL

When configuring LVS, you should always begin by configuring the primary router with the **Piranha Configuration Tool**. To do this,verify that the `piranha-gui` service is running and an administrative password has been set, as described in Sezione 2.2, «Impostazione password per il **Piranha Configuration Tool**».

If you are accessing the machine locally, you can open `http://localhost:3636` in a Web browser to access the **Piranha Configuration Tool**. Otherwise, type in the hostname or real IP address for the server followed by `:3636`. Once the browser connects, you will see the screen shown in Figura 4.1, «The Welcome Panel».

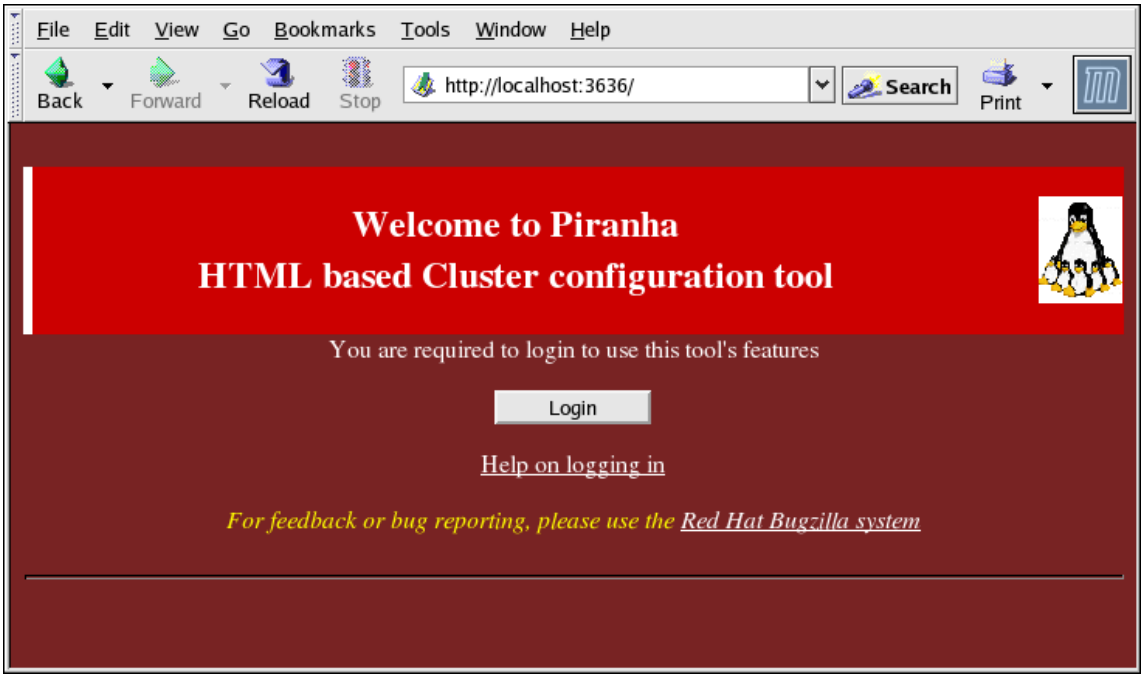


Figura 4.1. The Welcome Panel

Fate clic sul pannello **Login** ed inserite **piranha** per il **Nome utente**, e la password amministrativa da voi creata nel campo **Password**.

Il **Piranha Configuration Tool** è costituito da quattro schermate principali o *pannelli*. In aggiunta, il pannello **Server virtuali** contiene quattro *sottosezioni*. Il pannello **CONTROLLO/MONITORAGGIO** è il primo pannello dopo la schermata di registrazione.

4.3. CONTROL/MONITORING

Il pannello **CONTROLLO/MONITORAGGIO** presenta all'amministratore uno stato di runtime limitato di LVS. Esso contiene lo stato del demone **pulse**, la tabella per il routing LVS, ed i processi **nanny** generati da LVS.

NOTA

The fields for **CURRENT LVS ROUTING TABLE** and **CURRENT LVS PROCESSES** remain blank until you actually start LVS, as shown in Sezione 4.8, «Avvio di LVS».

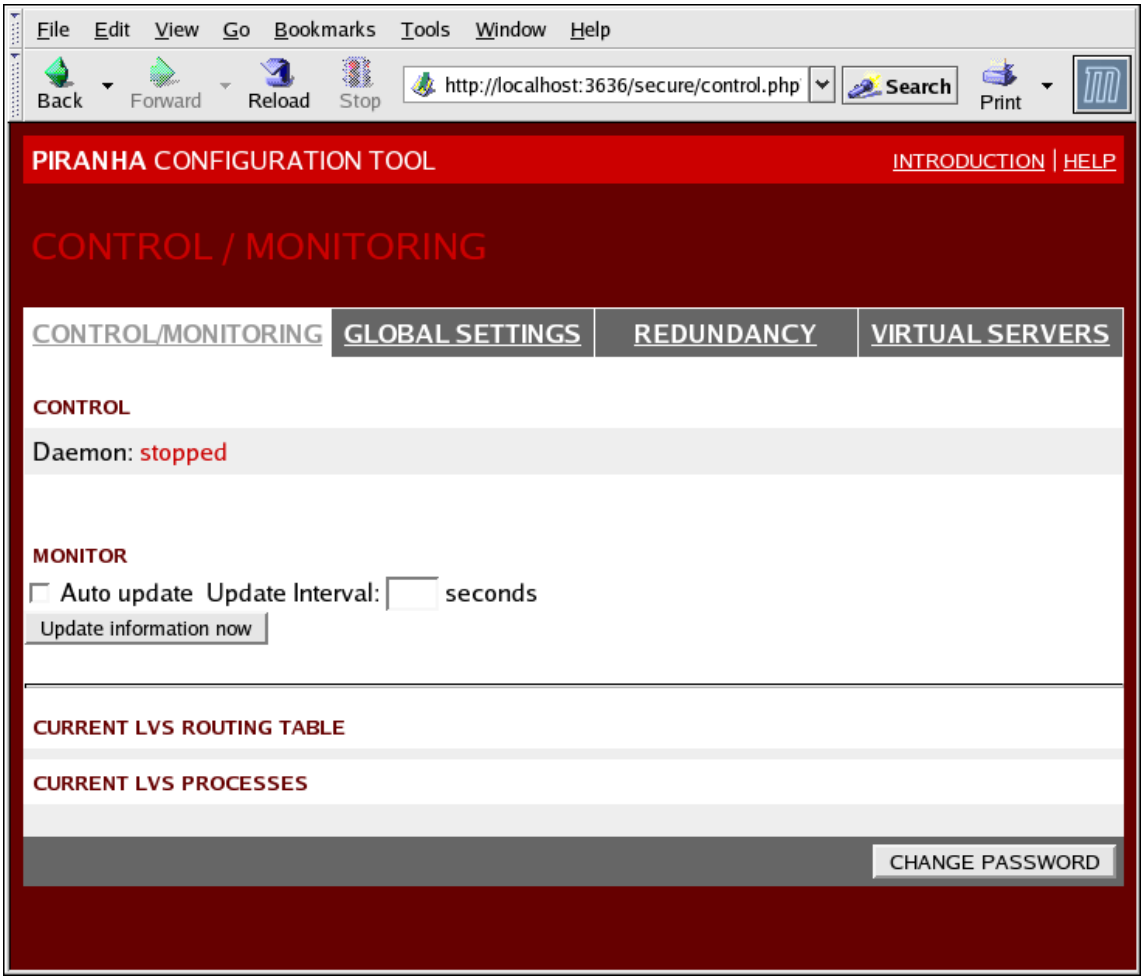


Figura 4.2. The CONTROL/MONITORING Panel

`Auto update`

Lo stato riportato in questa pagina può essere aggiornato automaticamente ad un intervallo configurato dall'utente. Per abilitare questa funzione fate clic sulla casella **`Auto update`**, ed impostate la frequenza di aggiornamento desiderata nella casella di testo **`Frequenza di aggiornamento in secondi`** (il valore di default è 10 secondi).

Non è consigliato impostare l'aggiornamento automatico ad un intervallo inferiore a 10 secondi. Se l'intervallo è inferiore a dieci secondi, sarà molto difficile riconfigurare l'intervallo **`Auto update`** poichè la pagina verrà aggiornata troppo frequentemente. Se incontrate questo problema, fate clic su di un pannello diverso per poi riselezionare **`CONTROLLO/MONITORAGGIO`**.

La funzione **`Auto update`** non funziona con tutti i browser, come ad esempio **Mozilla**.

`Update information now`

È possibile aggiornare manualmente le informazioni sullo stato facendo clic su questo pulsante.

`CHANGE PASSWORD`

Facendo clic su questo pulsante verrete direzionati sulla schermata d'aiuto, la quale contiene le informazioni utili per modificare la password amministrative per il **Piranha Configuration Tool**.

4.4. `GLOBAL SETTINGS`

The **GLOBAL SETTINGS** panel is where the you define the networking details for the primary LVS router's public and private network interfaces.

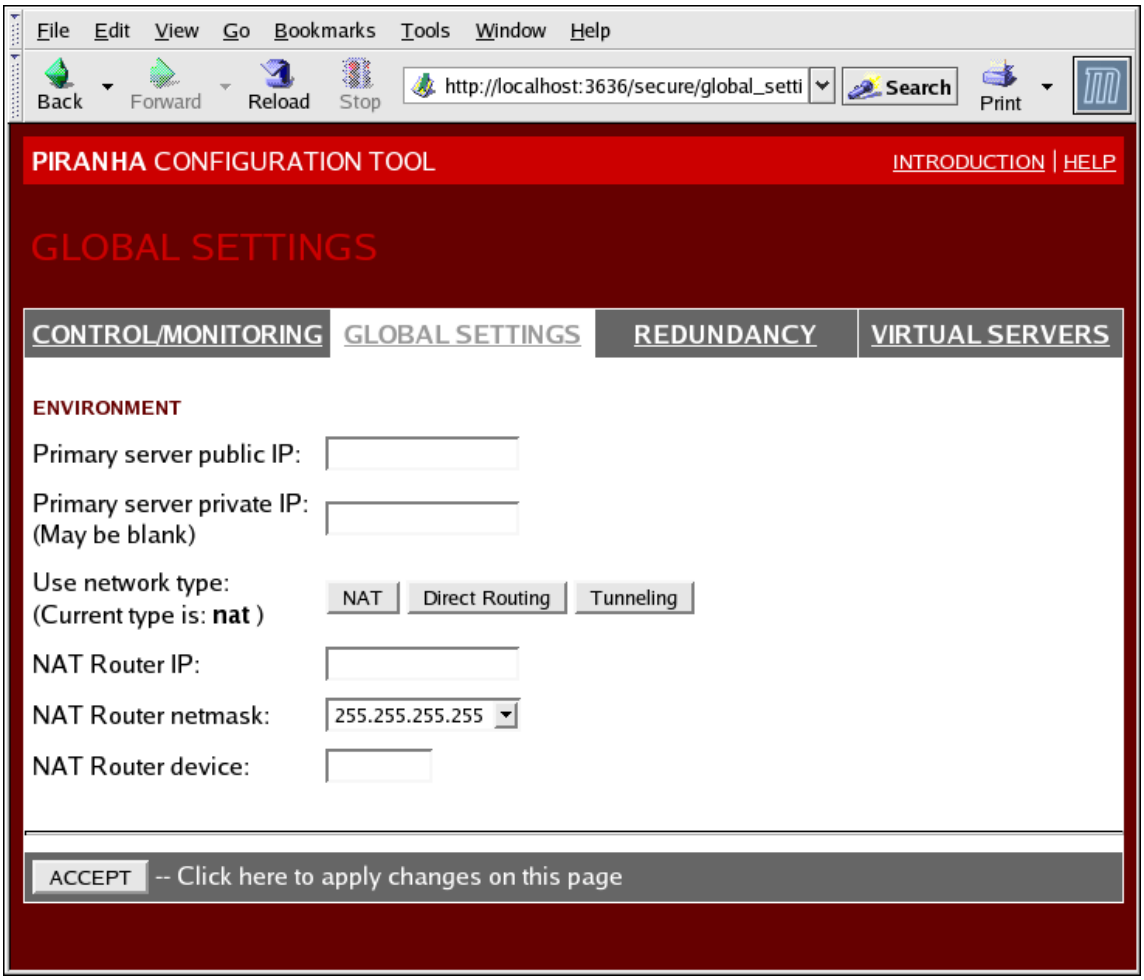


Figura 4.3. The GLOBAL SETTINGS Panel

The top half of this panel sets up the primary LVS router's public and private network interfaces. These are the interfaces already configured in Sezione 3.1.1, «Configurazione delle interfacce di rete per LVS con NAT».

Primary server public IP

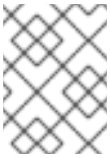
In questo campo inserite l'indirizzo IP reale instradabile pubblicamente per il nodo LVS primario.

Primary server private IP

Enter the real IP address for an alternative network interface on the primary LVS node. This address is used solely as an alternative heartbeat channel for the backup router and does not have to correlate to the real private IP address assigned in Sezione 3.1.1, «Configurazione delle interfacce di rete per LVS con NAT». You may leave this field blank, but doing so will mean there is no alternate heartbeat channel for the backup LVS router to use and therefore will create a single point of failure.

NOTA

L'indirizzo IP privato non è necessario per le configurazioni dell'**Instradamento diretto**, poichè tutti i real server e le directory LVS condividono gli stessi indirizzi IP virtuali, e dovrebbero avere la stessa configurazione IP route.



NOTA

The primary LVS router's private IP can be configured on any interface that accepts TCP/IP, whether it be an Ethernet adapter or a serial port.

`Use network type`

Fate clic su **`NAT`** per selezionare un NAT routing.

Fate clic su **`Instradamento diretto`** per selezionare l'instradamento diretto.

The next three fields deal specifically with the NAT router's virtual network interface connecting the private network with the real servers. These fields *do not* apply to the direct routing network type.

`NAT Router IP`

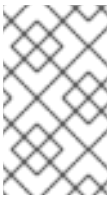
Inserite in questo campo l'IP floating privato. Il suddetto IP floating dovrebbe essere usato come gateway per i real server.

`NAT Router netmask`

If the NAT router's floating IP needs a particular netmask, select it from drop-down list.

`NAT Router device`

Usate questo campo per deifinire il nome del dispositivo dell'interfaccia di rete per l'indirizzo IP floating, come ad esempio **`eth1:1`**.



NOTA

Eseguite l'alias dell'indirizzo IP floating del NAT sull'interfaccia Ethernet collegata alla rete privata. In questo esempio la rete privata si trova sull'interfaccia **`eth1`**, e quindi **`eth1:1`** risulta essere l'indirizzo IP floating.

AVVERTIMENTO

Dopo aver completato questa pagina fate clic sul pulsante **ACCETTA**, in modo da non perdere le modifiche durante la selezione di un nuovo pannello.

4.5. REDUNDANCY

Il pannello **RIDONDANZA** vi permette di configurare il nodo del router LVS di backup, e d'impostare diverse opzioni per il monitoraggio dell'heartbeat.

NOTA

The first time you visit this screen, it displays an "inactive" **Backup** status and an **ENABLE** button. To configure the backup LVS router, click on the **ENABLE** button so that the screen matches Figura 4.4, «The **REDUNDANCY** Panel».

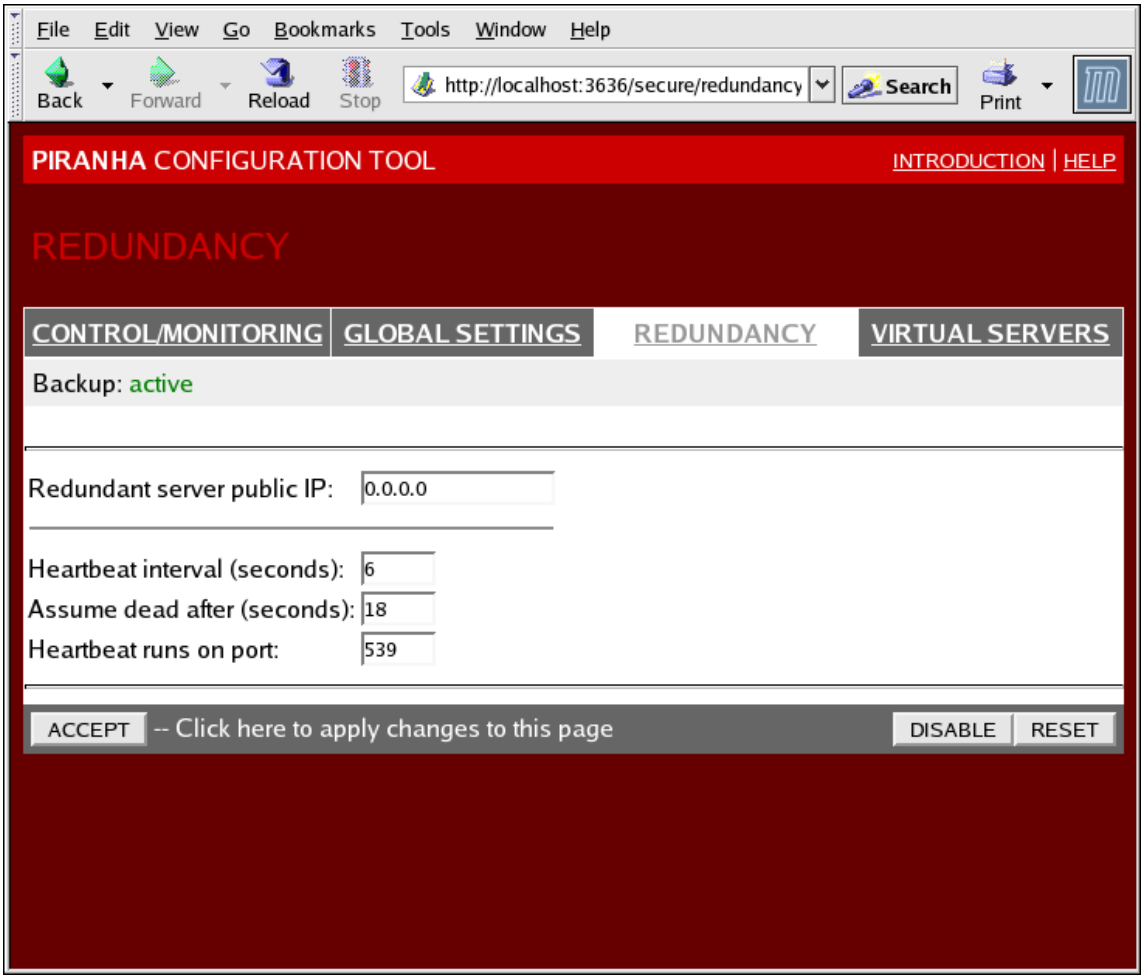


Figura 4.4. The REDUNDANCY Panel

`Redundant server public IP`

Inserite l'indirizzo IP reale pubblico per il nodo del router LVS di backup.

`Redundant server private IP`

Enter the backup node's private real IP address in this text field.

Se non siete in grado di visualizzare il campo chiamato **`IP privato del server ridondante`**, tornate sul pannello **`IMPOSTAZIONI GLOBALI`** ed inserite un indirizzo **`IP privato del server primario`** e successivamente fate clic su **`ACCETTA`**.

È possibile utilizzare il resto del pannello per configurare il canale heartbeat, il quale viene usato dal nodo di backup per controllare la presenza di errori nel nodo primario.

`Heartbeat Interval (seconds)`

Questo campo imposta il numero di secondi che intercorrono tra gli heartbeat — l'intervallo entro il quale il nodo di backup controllerà lo stato di funzionalità del nodo LVS primario.

Assume dead after (seconds)

Se il nodo LVS primario non risponde dopo questo intervallo in secondi, allora il nodo del router LVS di backup inizierà un processo di failover.

Heartbeat runs on port

Questo campo imposta la porta sulla quale l'heartbeat comunica con il nodo LVS primario. Il default è impostato su 539 se questo campo viene lasciato vuoto.

AVVERTIMENTO

Ricordate di selezionare il pulsante **ACCETTA** dopo aver eseguito qualsiasi modifica in questo pannello, in modo da non perdere le modifiche fatte durante la selezione di un nuovo pannello.

4.6. VIRTUAL SERVERS

Il pannello **SERVER VIRTUALI** visualizza le informazioni per ogni server virtuale attualmente definito. Ogni voce della tabella mostra lo stato del server virtuale, il nome del server, l'indirizzo IP assegnato al server, la maschera di rete dell'IP virtuale, il numero della porta attraverso la quale il servizio comunica, il protocollo usato e l'interfaccia del dispositivo virtuale.

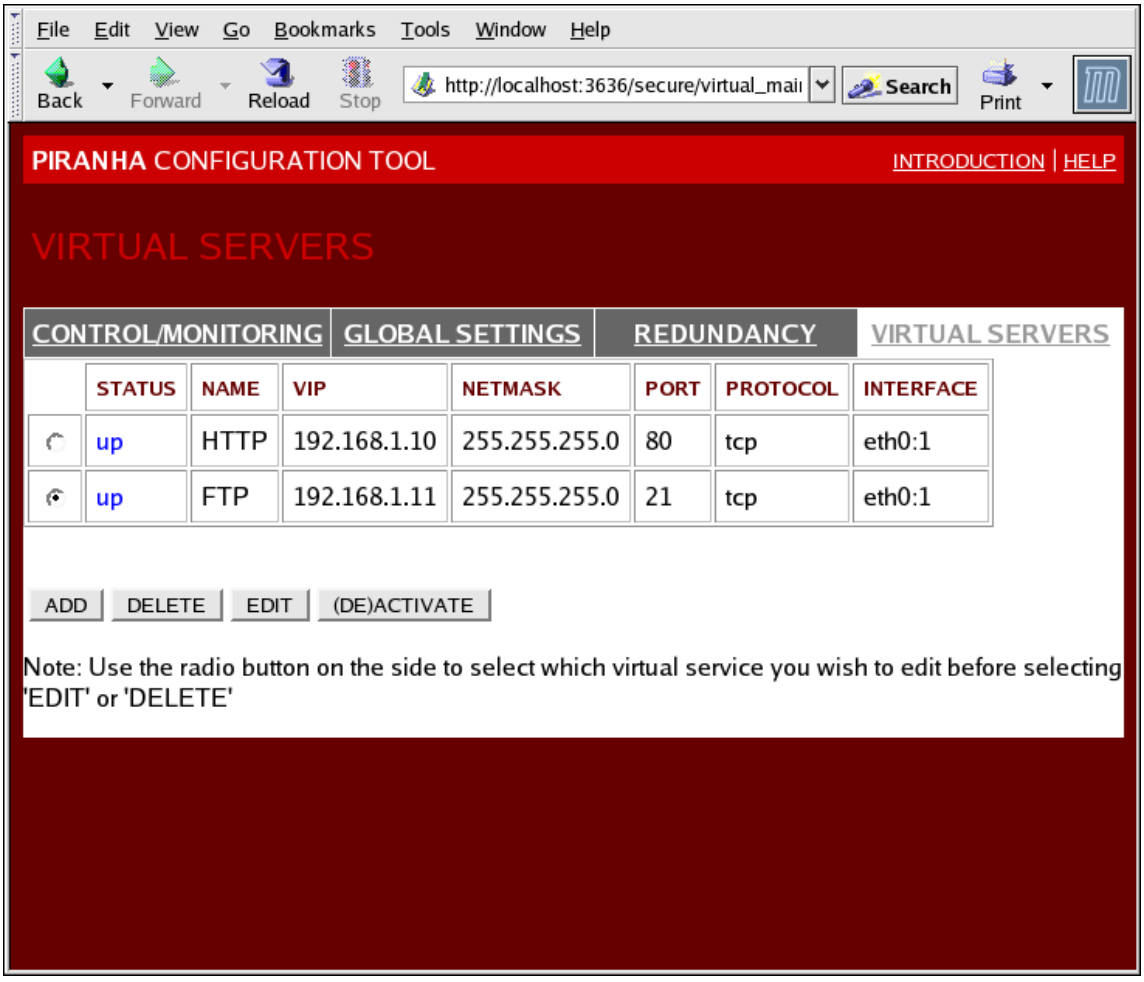


Figura 4.5. The VIRTUAL SERVERS Panel

Ogni server presente nel pannello **SERVER VIRTUALI** può essere configurato sulle sottoschermate seguenti o *sottosezioni*.

Per aggiungere un servizio fate clic su **AGGIUNGI**. Per rimuovere un servizio, selezionatelo attraverso il pulsante di selezione situato accanto al server virtuale, e successivamente fate clic su **CANCELLA**.

Per abilitare o disabilitare un server virtuale nella tabella, fate clic sul pulsante di selezione e successivamente su **(DIS)ATTIVA**.

Dopo aver aggiunto un server virtuale sarà possibile configurarlo tramite il pulsante di selezione situato sulla sinistra, e successivamente tramite la selezione del pulsante **MODIFICA** per visualizzare la sottosezione **SERVER VIRTUALE**.

4.6.1. La sottosezione SERVER VIRTUALE

The **VIRTUAL SERVER** subsection panel shown in Figura 4.6, «The VIRTUAL SERVERS Subsection» allows you to configure an individual virtual server. Links to subsections related specifically to this virtual server are located along the top of the page. But before configuring any of the subsections related to this virtual server, complete this page and click on the **ACCEPT** button.

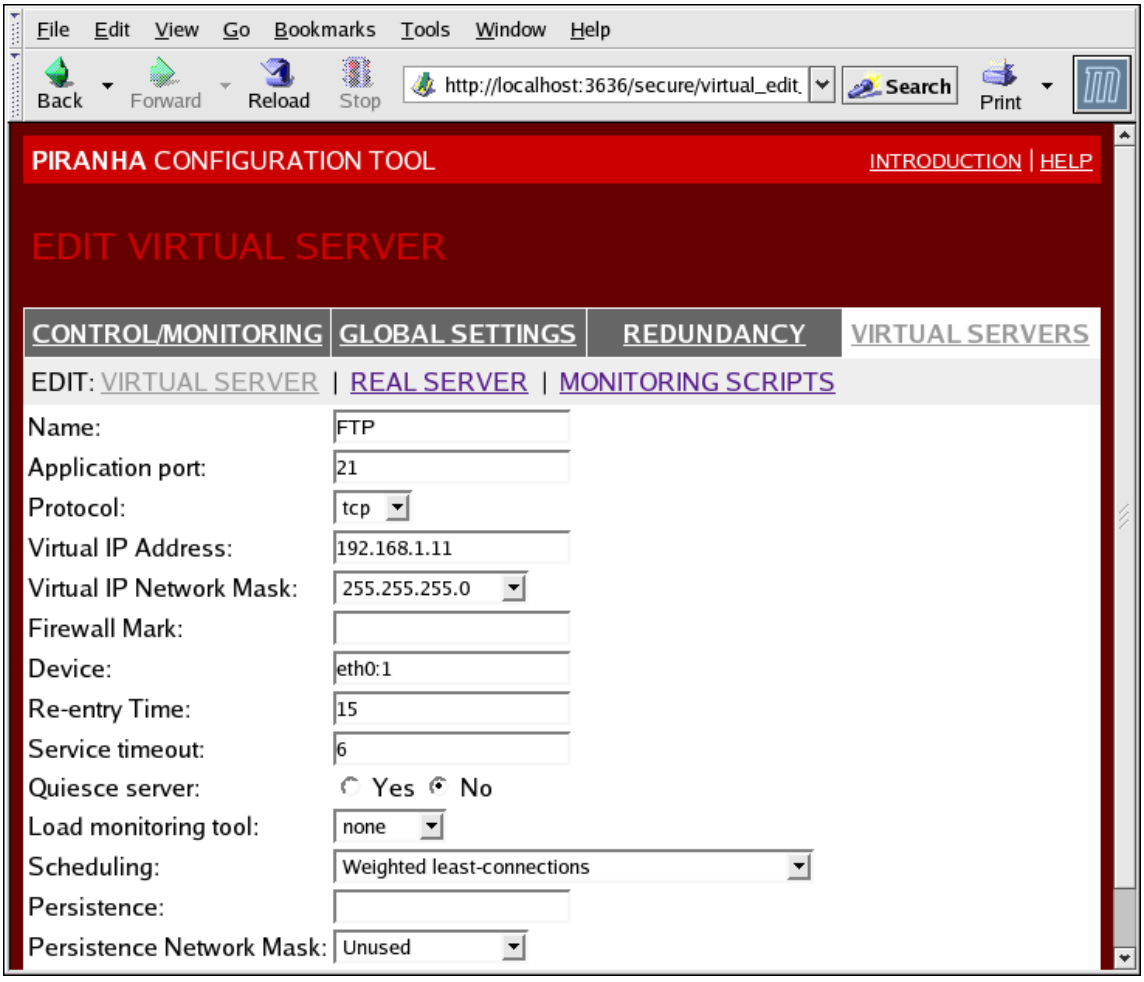


Figura 4.6. The VIRTUAL SERVERS Subsection

Name

Inserite un nome descrittivo per identificare il server virtuale. Questo nome *non* è l'hostname per la macchina, quindi rendetelo descrittivo e facilmente identificabile. È possibile far riferimento al protocollo usato dal server virtuale, come ad esempio HTTP.

Application port

Inserite il numero della porta attraverso la quale l'applicazione del servizio sarà in ascolto. Poichè questo esempio si riferisce ai servizi HTTP, la porta usata è 80.

Protocol

Scegliete tra UDP e TCP nel menù a tendina. I web server generalmente comunicano tramite il protocollo TCP, per questo motivo esso viene selezionato nell'esempio sopra riportato.

Virtual IP Address

Enter the virtual server's floating IP address in this text field.

Virtual IP Network Mask

Impostate la maschera di rete per questo server virtuale attraverso il menù a tendina.

Firewall Mark

Non inserite alcun valore intero per il firewall mark in questo campo, a meno che non raggruppate protocolli multi-port o create un server virtuale multi-port per protocolli separati ma correlati. In

questo esempio il server virtuale presenta un **Firewall Mark** di 80, poichè stiamo raggruppando i collegamenti per HTTP sulla porta 80 e per HTTPS sulla porta 443 utilizzando il valore 80 del firewall mark. Combinata con la persistenza questa tecnica è in grado di assicurare agli utenti che accedono pagine web sicure e non sicure, un instradamento allo stesso real server conservandone così lo stato.

AVVERTIMENTO

Entering a firewall mark in this field allows IPVS to recognize that packets bearing this firewall mark are treated the same, but you must perform further configuration outside of the **Piranha Configuration Tool** to actually assign the firewall marks. See Sezione 3.4, «Servizi multi-port e LVS» for instructions on creating multi-port services and Sezione 3.5, «Configurazione di FTP» for creating a highly available FTP virtual server.

Device

Inserite il nome del dispositivo di rete al quale desiderate legare l'indirizzo IP floating definito nel campo **Indirizzo IP virtuale**.

Eseguite l'alias dell'indirizzo IP floating pubblico sull'interfaccia Ethernet collegata alla rete pubblica. In questo esempio la rete pubblica si trova sull'interfaccia **eth0**, e quindi **eth0:1** deve essere inserito come nome del dispositivo.

Re-entry Time

Inserite un valore intero il quale definisce la durata del periodo, in secondi, prima che il router LVS attivo cerchi di ripristinare un real server nel gruppo dopo il verificarsi di un errore.

Service Timeout

Inserite un valore intero il quale definisce la durata del periodo, in secondi, prima che un real server venga considerato morto e quindi rimosso dal gruppo.

Quiesce server

Quando il pulsante di selezione del **server Quiesce** viene selezionato, ogni qualvolta un nodo del real server risulta online la tabella delle least-connection viene azzerata. In questo modo il router LVS attivo instrada le richieste come se tutti i real server sono stati appena aggiunti al cluster. Questa opzione impedisce ad un nuovo server di bloccarsi a causa di un elevato numero di collegamenti subito dopo l'ingresso in un gruppo.

Load monitoring tool

Il router LVS è in grado di monitorare il carico sui diversi real server attraverso l'utilizzo di **rup** o **ruptime**. Se **rup** viene selezionato dal menù a tendina, ogni real server deve eseguire il servizio **rstatd**. Se invece selezionate **ruptime**, ogni real server deve eseguire il servizio **rwhod**.

AVVERTIMENTO

Il controllo del carico *non* è sinonimo di bilanciamento del carico, e può risultare in un comportamento di schedulazione difficile da prevedere quando combinato con algoritmi di schedulazione 'pesati'. Altresì, se utilizzate il monitoraggio del carico i real server devono essere macchine Linux.

`Scheduling`

Select your preferred scheduling algorithm from the drop-down menu. The default is **`Weighted least-connection`**. For more information on scheduling algorithms, see Sezione 1.3.1, «Algoritmi di schedulazione».

`Persistence`

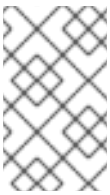
Se un amministratore necessita di collegamenti persistenti per il server virtuale durante le transazioni del client, inserite in questo campo il numero di secondi di inattività permessi prima dell'interruzione del collegamento.

IMPORTANTE

If you entered a value in the **`Firewall Mark`** field above, you should enter a value for persistence as well. Also, be sure that if you use firewall marks and persistence together, that the amount of persistence is the same for each virtual server with the firewall mark. For more on persistence and firewall marks, refer to Sezione 1.5, «Persistenza e Firewall mark».

Persistence Network Mask

Per limitare la persistenza su sottoreti particolari, selezionate la maschera di rete appropriata dal menù a tendina.



NOTA

Prima dell'avvento dei firewall mark la persistenza, limitata in base alle sottoreti, rappresentava un metodo artigianale per unificare i collegamenti. Ora per ottenere lo stesso risultato è consigliato utilizzare la persistenza in relazione ai firewall mark.

AVVERTIMENTO

Ricordate di selezionare il pulsante **ACCETTA** dopo aver eseguito qualsiasi cambiamento in questo pannello, per non perdere alcuna modifica durante la selezione di un nuovo pannello.

4.6.2. Sottosezione REAL SERVER

Facendo clic sul link **REAL SERVER** nella parte alta del pannello, verrà visualizzata la sottosezione **MODIFICA REAL SERVER**. Essa mostra lo stato degli host del server fisico per un servizio virtuale particolare.

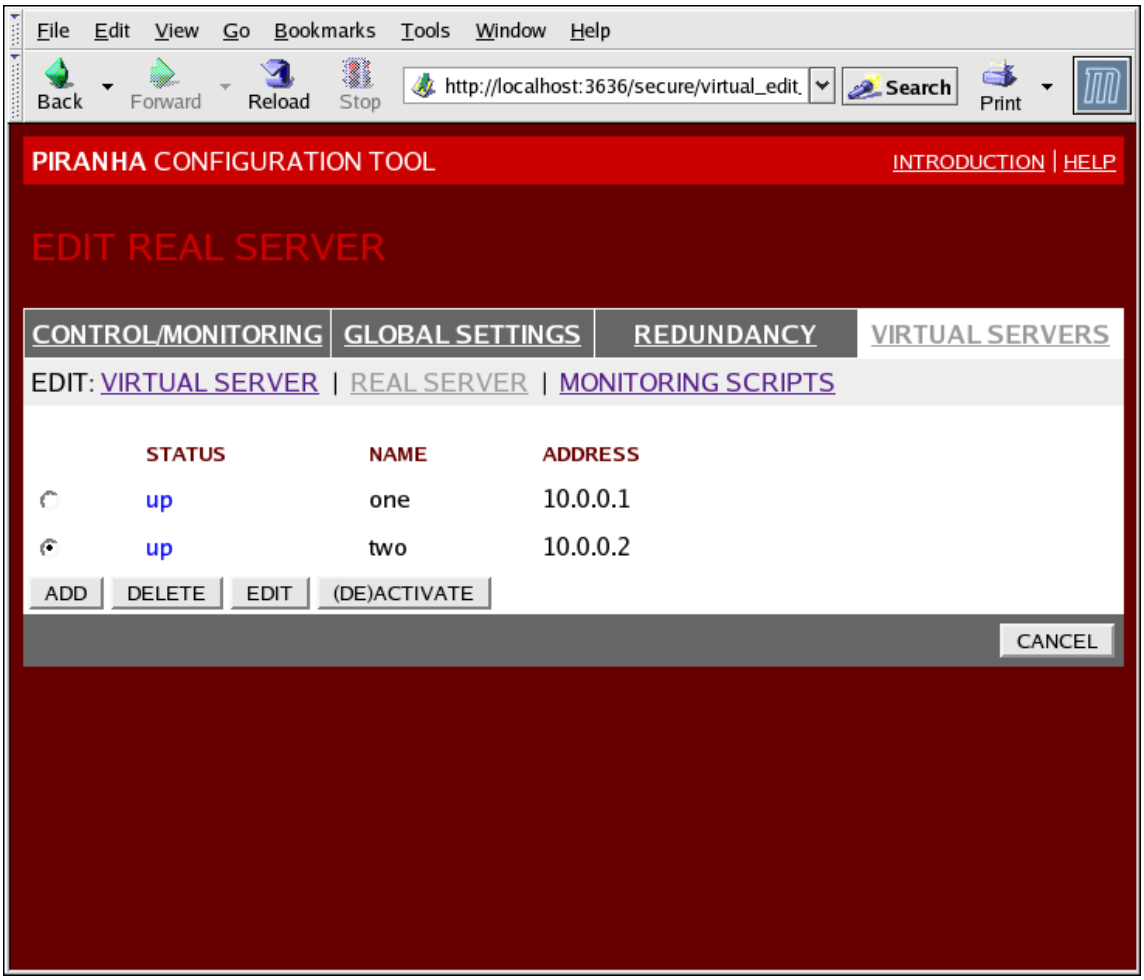


Figura 4.7. The REAL SERVER Subsection

Click the **ADD** button to add a new server. To delete an existing server, select the radio button beside it and click the **DELETE** button. Click the **EDIT** button to load the **EDIT REAL SERVER** panel, as seen in Figura 4.8, «The **REAL SERVER** Configuration Panel».

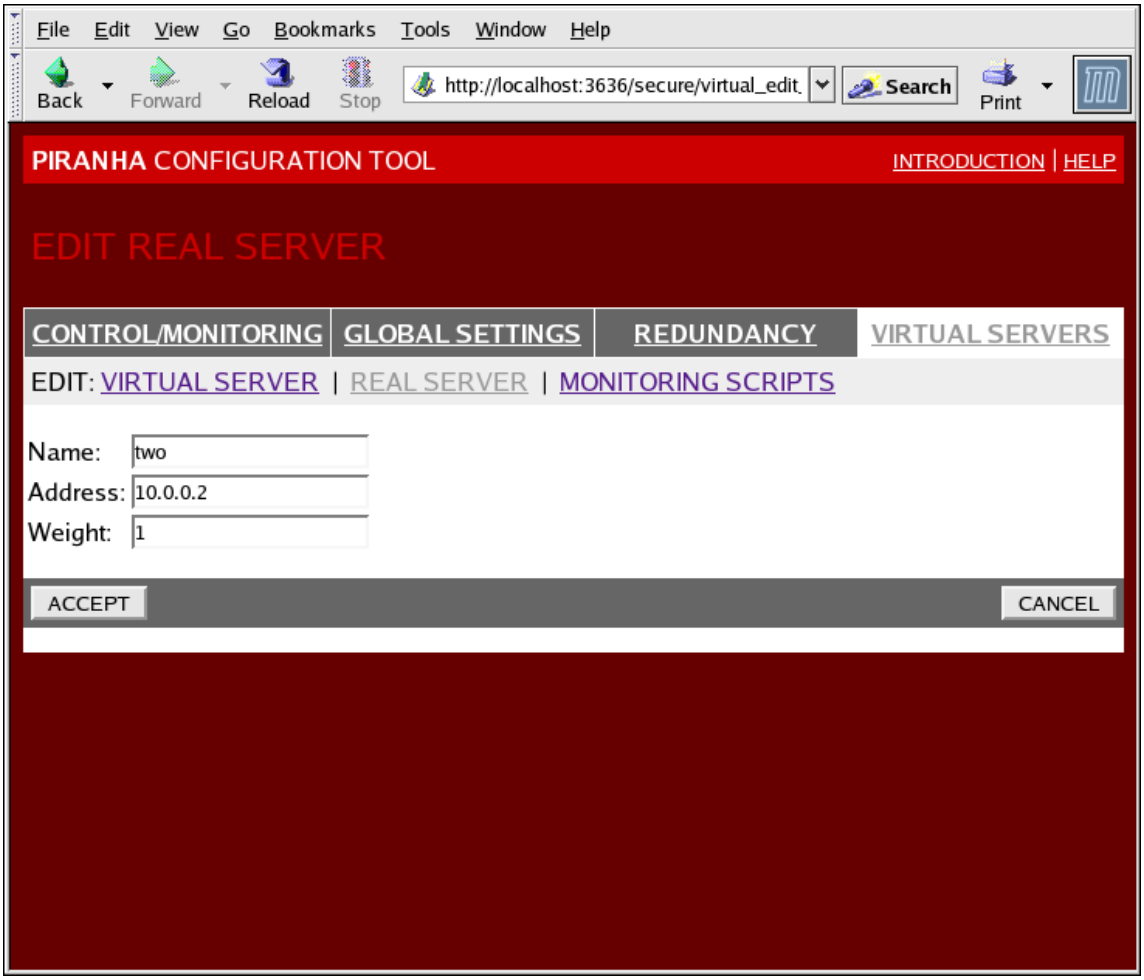
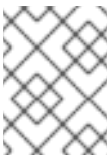


Figura 4.8. The REAL SERVER Configuration Panel

Questo pannello consiste in tre campi:

`Name`

Un nome descrittivo per il real server.



NOTA

Questo nome *non* è l'hostname per la macchina, quindi rendetelo descrittivo e facilmente identificabile.

`Address`

The real server's IP address. Since the listening port is already specified for the associated virtual server, do not add a port number.

`Weight`

An integer value indicating this host's capacity relative to that of other hosts in the pool. The value can be arbitrary, but treat it as a ratio in relation to other real servers in the pool. For more on server weight, see Sezione 1.3.2, «Peso del server e Schedulazione».

AVVERTIMENTO

Ricordate di selezionare il pulsante **ACCETTA** dopo aver eseguito qualsiasi modifica in questo pannello, in modo da non perdere le modifiche fatte durante la selezione di un nuovo pannello.

4.6.3. `EDIT MONITORING SCRIPTS` Subsection

Fate clic sul link **`SCRIPT DI MONITORAGGIO`** nella parte alta della pagina. La sottosezione **`MODIFICA SCRIPT DI MONITORAGGIO`** permette all'amministratore di specificare una sequenza della stringa send/expect, per verificare che il servizio per il server virtuale sia in funzione su ogni real server. Qui l'amministratore può anche specificare script personalizzati per controllare i servizi che richiedono una modifica dinamica dei dati.

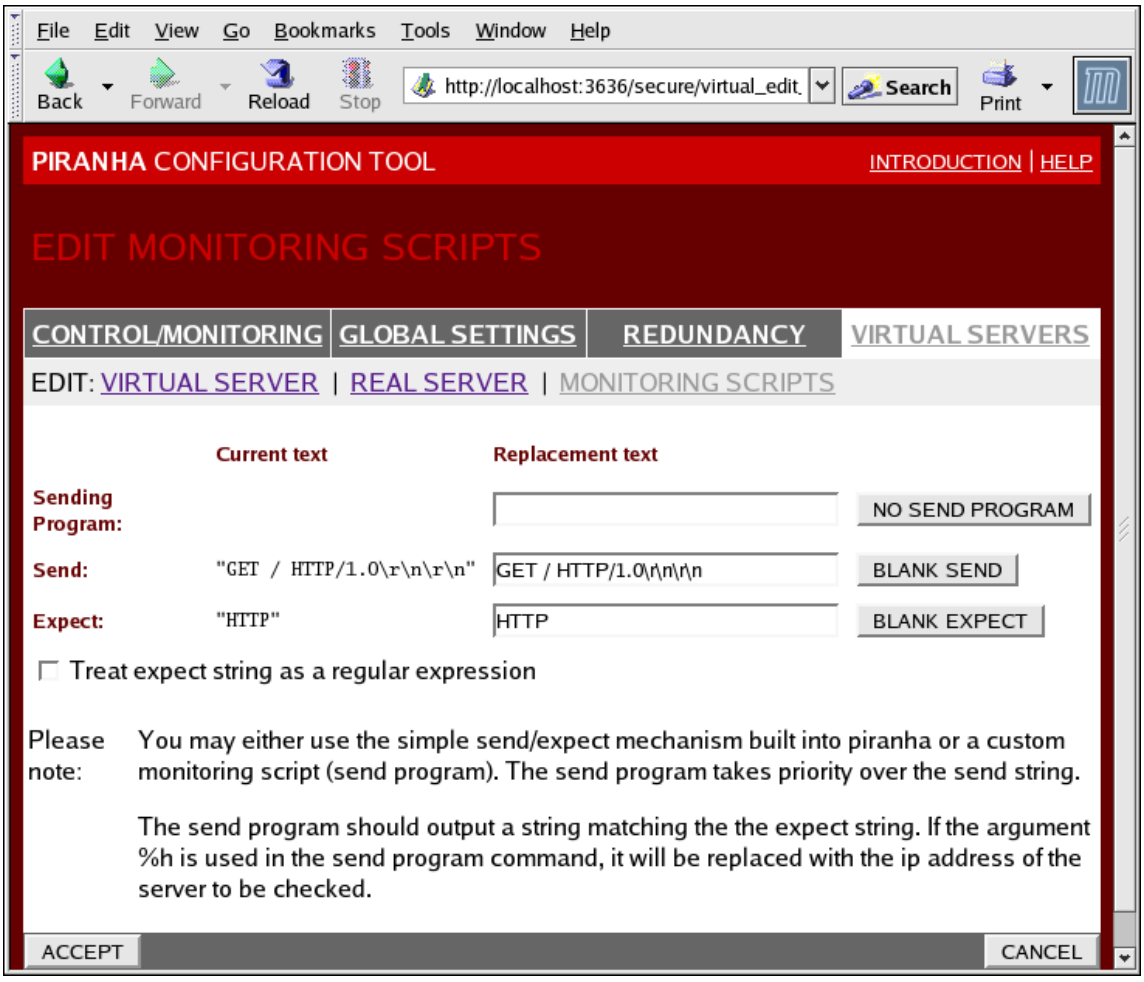
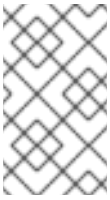


Figura 4.9. The `EDIT MONITORING SCRIPTS` Subsection

`Sending Program`

Per una verifica del servizio più avanzata è possibile utilizzare questo campo per specificare il percorso per uno script verifica-servizio. Questa funzionalità è molto utile per i servizi che richiedono la modifica dinamica dei dati, come ad esempio HTTPS o SSL.

Per utilizzare questa funzione è necessario scrivere uno script in grado di ritornare una risposta testuale, impostatelo per essere eseguibile e digitate il percorso nel campo **Programma mittente**.



NOTA

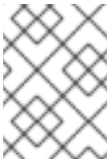
To ensure that each server in the real server pool is checked, use the special token **%h** after the path to the script in the **Sending Program** field. This token is replaced with each real server's IP address as the script is called by the **nanny** daemon.

Il seguente è un esempio di scirpt da utilizzare come guida durante la composizione di uno script esterno per il controllo-servizio:

```
#!/bin/sh

TEST=`dig -t soa example.com @$1 | grep -c dns.example.com

if [ $TEST != "1" ]; then
 echo "OK
else
 echo "FAIL"
fi
```



NOTA

Se viene inserito all'interno del campo **Programma mittente** un programma esterno, allora il campo **Invio** viene ignorato.

Send

Inserite in questo campo una stringa per il demone **nanny** da inviare ad ogni real server. Per default il campo invio viene completato per HTTP. Potrete alterare questo valore a seconda delle vostre necessità. Se lasciate questo campo vuoto, il demone **nanny** cercherà di aprire la porta, e se avrà successo, assumerà che il servizio sia in esecuzione.

In questo campo è permesso solo una sequenza di invio, e può contenere solo caratteri stampabili, ASCII, insieme ai seguenti caratteri di escape:

- \n per nuove righe.
- \r per il ritorno a capo del cursore.
- \t per tab.
- \ per eseguire l'escape del carattere successivo.

Expect

Inserite una risposta testuale che il server deve ritornare se funziona correttamente. Se avete compilato il vostro programma mittente, inserite la risposta da inviare in caso di successo.

NOTA

Per determinare cosa inviare per un determinato servizio, aprite un collegamento **`telnet`** per la porta su di un real server, e controllate cosa viene ritornato. Per esempio FTP riporta 220 previo collegamento, è possibile quindi inserire **`quit`** nel campo **`Invia`** e **`220`** nel campo **`In attesa`**.

AVVERTIMENTO

Ricordate di selezionare il pulsante **ACCETTA** dopo aver eseguito qualsiasi modifica in questo pannello, in modo da non perdere le modifiche fatte durante la selezione di un nuovo pannello.

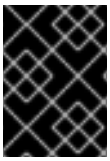
Once you have configured virtual servers using the **Piranha Configuration Tool**, you must copy specific configuration files to the backup LVS router. See Sezione 4.7, «Sincronizzazione dei file di configurazione» for details.

4.7. SINCRONIZZAZIONE DEI FILE DI CONFIGURAZIONE

Dopo aver configurato il router LVS primario, sono presenti diversi file di configurazione da copiare sul router LVS di backup prima di avviare LVS.

Questi file includono:

- **`/etc/sysconfig/ha/lvs.cf`** – il file di configurazione per i router LVS.
- **`/etc/sysctl`** – il file di configurazione che, oltre ad altre funzioni, abilita il packet forwarding nel kernel.
- **`/etc/sysconfig/iptables`** – Se state utilizzando i firewall mark, dovreste sincronizzare uno di questi file in base al filtro del pacchetto di rete da voi utilizzato.



IMPORTANTE

I file **`/etc/sysctl.conf`** e **`/etc/sysconfig/iptables`** *non* cambiano quando configurate LVS utilizzando il **Piranha Configuration Tool**.

4.7.1. Sincronizzazione di `lvs.cf`

Ogni qualvolta il file di configurazione LVS, **`/etc/sysconfig/ha/lvs.cf`**, viene aggiornato o creato, sarà necessario copiarlo sul nodo del router LVS di backup.

AVVERTIMENTO

Sia il nodo del router LVS attivo che quello passivo devono avere file **lvs.cf** identici. Se i file di configurazione LVS dei nodi del router LVS non corrispondono, il processo di failover non si potrà verificare.

Il modo migliore per fare ciò è quello di utilizzare il comando **scp**.

IMPORTANTE

To use **scp** the **sshd** must be running on the backup router, see Sezione 2.1, «Configurazione dei servizi sui router LVS» for details on how to properly configure the necessary services on the LVS routers.

Emettete il seguente comando come utente root dal router LVS primario per sincronizzare i file **lvs.cf** tra i nodi del router:

```
scp /etc/sysconfig/ha/lvs.cf n.n.n.n:/etc/sysconfig/ha/lvs.cf
```

Nel comando sostituite *n.n.n.n* con l'indirizzo IP del real server del router LVS di backup.

4.7.2. Sincronizzazione di sysctl

Nella maggior parte delle situazioni il file **sysctl** viene modificato solo una volta. Questo file viene letto al momento dell'avvio ed indica al kernel di abilitare il packet forwarding.

IMPORTANTE

If you are not sure whether or not packet forwarding is enabled in the kernel, see Sezione 2.5, «Abilitazione del Packet Forwarding» for instructions on how to check and, if necessary, enable this key functionality.

4.7.3. Sincronizzazione delle regole di filtraggio dei pacchetti di rete

Se state utilizzando **iptables**, avrete bisogno di sincronizzare il file di configurazione appropriato sul router LVS di backup.

Se alterate qualsiasi regola del filtro del pacchetto di rete, inserite il seguente comando come utente root dal router LVS primario:

```
scp /etc/sysconfig/iptables n.n.n.n:/etc/sysconfig/
```

Nel comando sostituite *n.n.n.n* con l'indirizzo IP del real server del router LVS di backup.

Successivamente aprite una sessione **ssh** per il router di backup, oppure eseguite la registrazione sulla macchina come utente root e digitate il seguente comando:

```
/sbin/service iptables restart
```

Once you have copied these files over to the backup router and started the appropriate services (see Sezione 2.1, «Configurazione dei servizi sui router LVS» for more on this topic) you are ready to start LVS.

4.8. AVVIO DI LVS

Per avviare LVS è consigliato avere due terminali root aperti simultaneamente, oppure due sessioni **ssh** root per il router LVS primario.

In un terminale controllate i messaggi di log del kernel con il comando:

```
tail -f /var/log/messages
```

Successivamente avviate LVS digitando il seguente comando all'interno dell'altro terminale:

```
/sbin/service pulse start
```

Follow the progress of the **pulse** service's startup in the terminal with the kernel log messages. When you see the following output, the pulse daemon has started properly:

```
gratuitous lvs arps finished
```

Per non controllare più **/var/log/messages**, digitate **Ctrl+c**.

Da qui in avanti il router LVS primario è anche il router LVS attivo. A questo punto anche se sarà possibile effettuare richieste a LVS, è consigliato avviare il router LVS di backup prima d'innescare LVS. Per fare questo ripetete il processo sopra descritto sul nodo del router LVS di backup.

Dopo aver completato la fase finale, LVS sarà in esecuzione.

APPENDICE A. UTILIZZO DI LVS CON IL RED HAT CLUSTER

È possibile utilizzare i router LVS con Red Hat Cluster per implementare un sito high-availability e-commerce in grado di fornire un bilanciamento del carico, una integrità dei dati ed una disponibilità delle applicazioni.

The configuration in Figura A.1, «LVS with a Red Hat Cluster» represents an e-commerce site used for online merchandise ordering through a URL. Client requests to the URL pass through the firewall to the active LVS load-balancing router, which then forwards the requests to one of the Web servers. The Red Hat Cluster nodes serve dynamic data to the Web servers, which forward the data to the requesting client.

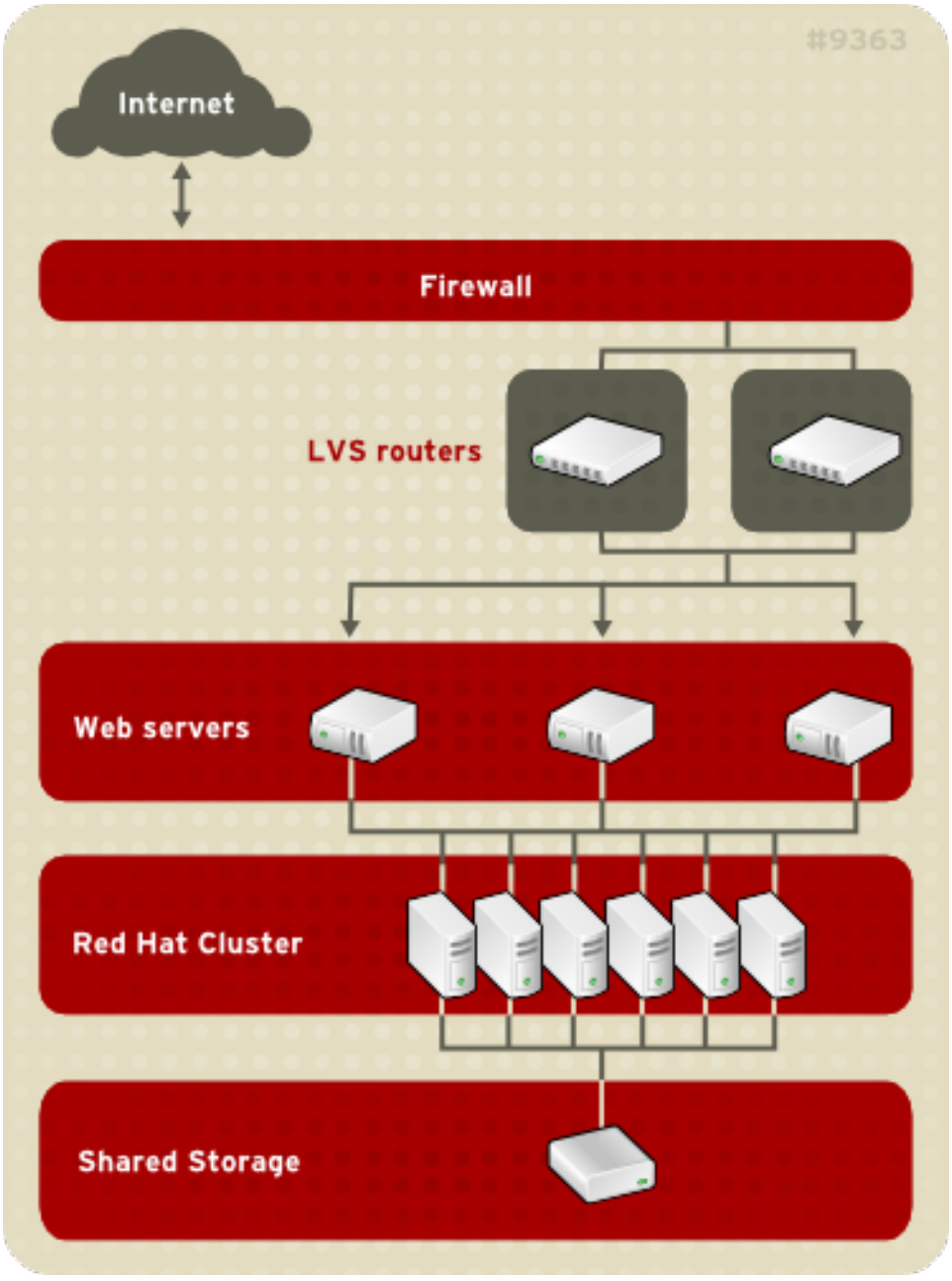


Figura A.1. LVS with a Red Hat Cluster

Serving dynamic Web content with LVS requires a three-tier configuration (as shown in Figura A.1, «LVS with a Red Hat Cluster»). This combination of LVS and Red Hat Cluster allows for the configuration of a high-integrity, no-single-point-of-failure e-commerce site. The Red Hat Cluster can run a high-availability instance of a database or a set of databases that are network-accessible to the Web servers.

Per fornire un contenuto dinamico è necessario avere una configurazione a tre livelli. Anche se la configurazione a due livelli è idonea se i web server servono solo contenuti web statici (i quali consistono in una quantità piccola di dati non frequentemente modificati), essa non è idonea se i web server servono un contenuto dinamico. Tale contenuto include un inventario dei prodotti, ordinazioni, o database dei clienti, i quali devono essere aggiornati e conformi su tutti i web server, in modo da assicurare ai clienti stessi un accesso a informazioni accurate e costantemente aggiornate.

Ogni livello fornisce le seguenti funzioni:

- Primo livello — i router LVS eseguono un bilanciamento del carico per distribuire le richieste web.
- Secondo livello — Un set di web server per servire le richieste.
- Terzo livello — Un Red Hat Cluster per servire i dati ai Web server.

In an LVS configuration like the one in Figura A.1, «LVS with a Red Hat Cluster» , client systems issue requests on the World Wide Web. For security reasons, these requests enter a Web site through a firewall, which can be a Linux system serving in that capacity or a dedicated firewall device. For redundancy, you can configure firewall devices in a failover configuration. Behind the firewall are LVS load-balancing routers, which can be configured in an active-standby mode. The active load-balancing router forwards the requests to the set of Web servers.

Each Web server can independently process an HTTP request from a client and send the response back to the client. LVS enables you to expand a Web site's capacity by adding Web servers behind the LVS routers; the LVS routers perform load balancing across a wider set of Web servers. In addition, if a Web server fails, it can be removed; LVS continues to perform load balancing across a smaller set of Web servers.

APPENDICE B. REVISION HISTORY

Revisione 5-8.400 **2013-10-31** **Rüdiger Landmann**

Rebuild with publican 4.0.0

Revisione 5-8 **2012-07-18** **Anthony Towns**

Rebuild for Publican 3.0

Revisione 2.0-0 **Mon Feb 08 2010** **Paul Kennedy**

Resolves: 492000

Changes -d to -s in arptables "OUT" directive in "Direct Routing and arptables_jf" section.

Revisione 1.0-0 **Tue Jan 20 2009** **Paul Kennedy**

Consolidation of point releases

INDICE ANALITICO

Simboli

A

C

D

F

I

J

L

M

N

P

R

S

W