

ALMA MATER STUDIORUM · UNIVERSITÀ DI
BOLOGNA

SCUOLA DI SCIENZE
Corso di Laurea in Informatica

VsnLib:
una interfaccia unica di
configurazione
per stack di rete virtuali tramite
pacchetti netlink

Relatore:
Chiar.mo Prof.
Renzo Davoli

Presentata da:
Simone Preite

Sessione di Dicembre
Anno Accademico 2016-2017

*Questa è la DEDICA:
ognuno può scrivere quello che vuole,
anche nulla . . .*

Introduzione

Nell'ambito delle reti negli ultimi anni con l'avvento di internet si è assistito ad una crescita esponenziale delle apparecchiature connesse a questa rete. Fino ad oggi questa immensa rete è stata gestita per lo più attraverso indirizzamento delle interfacce di rete di cui i computer sono dotati.

Al momento però ci si trova a fare i conti con la carenza degli indirizzi e pian piano si procede verso una migrazione basata sul nuovo (ma non troppo) protocollo IPv6 che prevede uno spazio di indirizzamento molto più ampio che lascia spazio a nuove idee impraticabili con protocollo IPv4. In seguito sentiremo parlare di IoT (Internet of Things) ed IoTh (Internet of Threads), due paradigmi che possiamo affermare essere il risultato di un'evoluzione sempre crescente dei sistemi informativi.

Indice

Introduzione	i
1 Internet of Threads	1
1.1 Paradigma	1
1.2 Stack Famosi	2
1.2.1 LWIP	2
1.2.2 LWIPv6	2
1.2.3 PicoTCP	2
2 VsnLib	3
2.1 Il Progetto	3
2.2 Sviluppo	4
2.3 Moduli	5
2.4 Il Futuro	5
3 Casi d’uso	7
3.1 Esempi	7
Conclusioni	9

A Prima Appendice	11
B Seconda Appendice	13
Bibliografia	15

Elenco delle figure

2.1	mappa concettuale libreria	4
-----	--------------------------------------	---

Elenco delle tabelle

Capitolo 1

Internet of Threads

1.1 Paradigma

IoT (Internet of Things), ovvero l'internet delle cose.

Questo concetto vuole rappresntare la diffusione dei sistemi embedded come veri e propri nodi di rete, possiamo definirlo il precursore del sistema che stiamo per descrivere ed è il sistema che ad oggi ha permesso di incontrare il nostro condizionatore o la nostra caldaia, ma addirittura il nostro tostapane, in rete.

Ad un certo punto si è sentita l'esigenza di elevare questa astrazione, nasce il concetto di IoTh (ovvero Internet of Threads).

L'idea diventa quella di avere processi come nodi di rete e non oggetti fisici, l'analogia è molto simile alla differenza tra telefoni fissi e telefoni cellulari, ovvero in passata era necessario pensare al luogo in cui una persona potesse trovarsi,, mentre attraverso l'assegnamento di un numero personale oggi possiamo comunicare direttamente con la persona desiderata[1, 2].

In termini di internet questo si traduce nella possibilità di migrare servizi

da un capo all'altro del mondo con la semplicità di un kill and start. Una nota va anche fatta in termini di sicurezza, ogni servizio può essere eseguito con il proprio stack di rete e questo impedirebbe a software di port mapping di carpire informazioni sulla macchina che ospita detto servizio. Inoltre possedendo il proprio stack di rete possiamo eseguire i processi con un utente non privilegiato e pertanto anche un bug del demone non comprometterebbe l'intero sistema.

1.2 Stack Famosi

Diversi sono i progetti che si occupano di offrire ai processi il proprio stack di rete; tra questi ne verranno presi in esame tre, considerati più indicati per uno studio in quanto open source.

- primo oggetto
- secondo oggetto

1.2.1 LWIP

1.2.2 LWIPv6

1.2.3 PicoTCP

Capitolo 2

VsnLib

Strato di compatibilit  tra pacchetti netlink e stack di rete virtuali.

2.1 Il Progetto

Gli stack analizzati in precedenza offrono a grandi linee la stessa tipologia di servizio seppur ognuna con le proprie caratteristiche.

Nessuno dei progetti ha per  tenuto in considerazione l'idea di utilizzare un'interfaccia di configurazione che fosse standard e pertanto   necessario usare le funzioni specifiche per ognuno di questi progetti, costringendo il programmatore a cambiare modalit  di interazione da stack a stack.

Il progetto di creare questa libreria nasce proprio da questa esigenza, ovvero cercare di uniformare le interfacce di comunicazione in modo tale che l'utente non sia costretto ad adattarsi ogni qual volta decida di cambiare stack.

2.2 Sviluppo

Come possiamo individuare nella successiva immagine, il progetto si compone di un core che comprende una parte client ed una server.

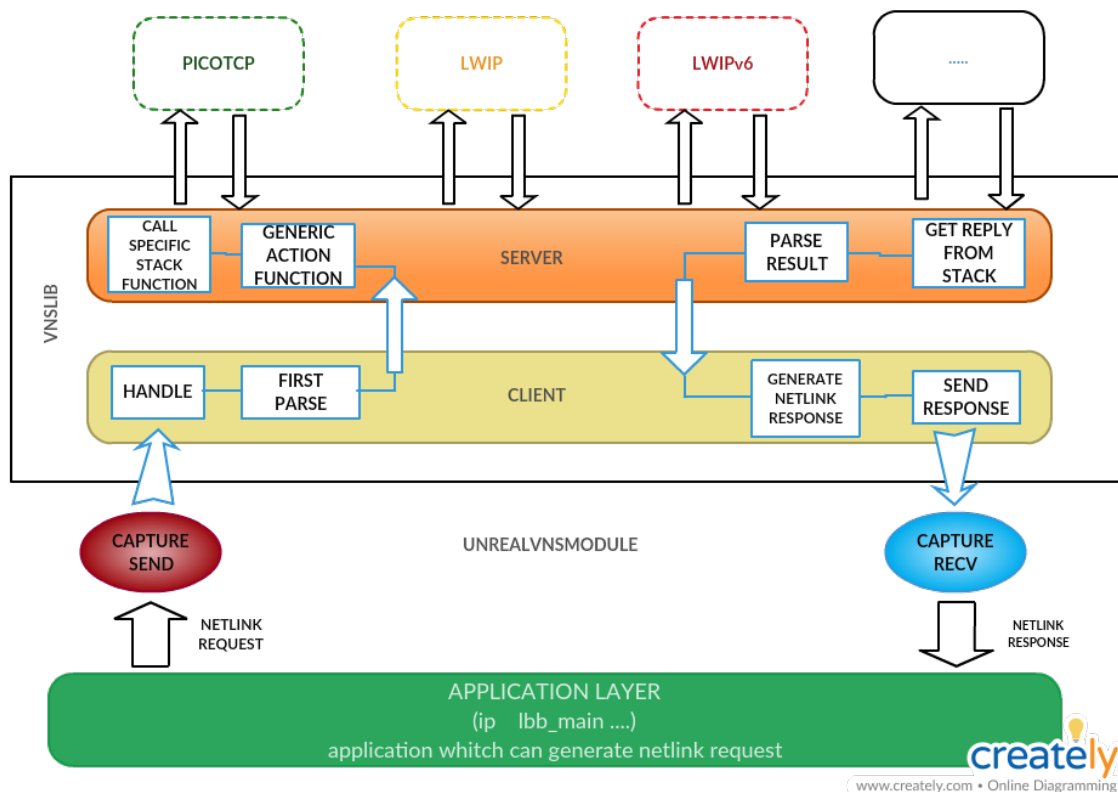


Figura 2.1: mappa concettuale libreria

È facile immaginare che la suddivisione riguardi l'interazione della libreria con l'utente.

Client-side: risiedono le funzioni che catturano i pacchetti netlink e che quindi permettono le configurazioni attraverso l'interfaccia netlink.

In questa fase ci si occupa di fare parsing del pacchetto per costruire una struttura contenente i dati significativi.

Server-side: back end della libreria, qui vengono gestiti i dati passati dal client per poi chiamare le specifiche funzioni approntate per configurare lo stack in uso.

2.3 Moduli

2.4 Il Futuro

Capitolo 3

Casi d'uso

Di seguito alcuni esempi di utilizzo della libreria. Le dimostrazioni seguenti sono state effettuate su una macchina con architettura amd64 e con installato il sistema operativo debian in versione sid (quindi unstable) ma sono state testate e riprodotte anche su altre configurazioni e distribuzioni GNU/Linux.

3.1 Esempi

Conclusioni

L'utilizzo di queste tipologie di stack dipende strettamente dalla diffusione del protocollo IPv6 e dei sistemi embedded e sembra essere la direzione in cui questo settore sta muovendosi. La ricerca e lo sviluppo di questi meccanismi sono necessari ad ottenere un sistema performante e stabile quando IPv6 sarà l'ordinario, vanno poi considerati anche i vantaggi conseguenti nelle reti virtuali che già ora vengono utilizzate per la sperimentazione.

L'IoTh ne è un'astrazione più forte. In queste conclusioni voglio fare un riferimento alla bibliografia: questo è il mio riferimento [3, 4].

Appendice A

Prima Appendice

In questa Appendice non si è utilizzato il comando:
`\clearpage{\pagestyle{empty}\cleardoublepage}`, ed infatti l'ultima pagina 8 ha l'intestazione con il numero di pagina in alto.

Appendice B

Seconda Appendice

Bibliografia

- [1] Renzo Davoli. Internet of Threads. <http://www.cs.unibo.it/~renzo/papers/2013.iciw.pdf>, 2013.
- [2] Renzo Davoli. Internet of Threads: Processes as Internet Nodes. <http://www.cs.unibo.it/~renzo/papers/2014.IntTechIoTh.pdf>, 2014.
- [3] Renzo Davoli. Internet of Threads. http://www.cs.unibo.it/~renzo/papers/ConfGARR11_SelectedPapers_Davoli.pdf.
- [4] Altran. picoTCP. <https://github.com/tass-belgium/picotcp>.
- [5] Virtual Square Team. vuos. <https://github.com/virtualsquare/vuos>.
- [6] Virtual Square Team. LWIPv6. <http://wiki.v2.cs.unibo.it/wiki/index.php/LWIPV6>.
- [7] Virtual Square Team. UMview. <http://wiki.v2.cs.unibo.it/wiki/index.php/UMview>.
- [8] Virtual Square Team. purelibc. <http://wiki.virtualsquare.org/wiki/index.php/PureLibc>.

Ringraziamenti

Qui possiamo ringraziare il mondo intero!!!!!!!!!!
Ovviamente solo se uno vuole, non è obbligatorio.