Internal Report INAF/IASF–BO 550/2009

SERVER DEDICATI ALLA MISSIONE PLANCK PRESSO IL CENTRO DI CALCOLO IASFBO

E. FRANCESCHI, A. DE ROSA

INAF / IASF – Bologna

11 dicembre 2009

SERVER DEDICATI ALLA MISSIONE PLANCK PRESSO IL CENTRO DI CALCOLO IASFBO

E. Franceschi, A. De Rosa, INAF / IASF – Bologna

Abstract

Questo documento vuole dare una rapida ma comunque sufficientemente esaustiva descrizione della piccola sottorete presente all'interno del Centro di Calcolo dell'Istituto di Astrofisica Spaziale e Fisica Cosmica di Bologna, riservata all'uso da parte della Comunità Scientifica di Planck/LFI. Ne vengono esaminati struttura, caratteristiche e potenzialità; i servizi implementati e le modalità con cui vengono resi disponibili; le modifiche e integrazioni previste nel breve termine; come anche le possibili interazioni con altre subnet, d'Istituto e non, con le problematiche di sicurezza che ne conseguono.

1. INTRODUZIONE

Intorno agli ultimi mesi del 2004, con l'avvicinarsi delle prime fasi operative di test sui componenti dello strumento LFI[1], divenne evidente la necessità di riorganizzare le risorse informatiche comuni accessibili in modo esclusivo al gruppo di ricercatori e tecnici coinvolti nel progetto Planck/LFI. L'unica vera risorsa comune installata, all'epoca, era una piccola workstation denominata "planck", con sistema operativo Windows 2000 Server e l'unica particolarità di avere uno storage di tipo SCSI; qualcosa insomma di già relativamente obsoleto.
Il primo provvedimento è stato quindi quello di sostituire detta workstation con un vero e proprio server – poi denominato "max" – da collocarsi all'interno del Centro di Calcolo, assieme alle altre risorse di calcolo e storage comuni agli altri gruppi di lavoro presenti all'interno dell'istituto IASFBO. L'accesso a *max* doveva comunque esser vincolato a controlli specifici, in un'ottica di necessaria riservatezza sui dati che vi sarebbero stati immagazzinati. Sin da subito poi, secondo la stessa logica, è stata evidenziata l'assoluta necessità di render *max* particolarmente "sicuro" – termine da valutarsi qui in relazione alla relativa "insicurezza" che caratterizzava i server d'Istituto all'epoca.
Dal lato dei servizi svolti, *max* doveva fornire un'area di storage dotata di un discreto livello di *fault tolerance*, accessibile sia via ftp che via ssh, a seconda dei dati considerati: solo ftp per i dati comuni, relativi alle campagne di calibrazione che sarebbero partite di lì a poco; ssh, come anche opzionalmente ftp, per i dati specifici di ogni utente. *Max* avrebbe difatti dovuto anche offrire ai suoi utenti un'area di storage riservata, da considerarsi come piccola area di backup personalizzata, in aggiunta alla area "`home`" già disponibile sui server d'Istituto per tutti gli utenti. E così in effetti è stato, anche se con un utilizzo meno generalizzato di quanto fosse stato previsto.
Un'altra funzionalità contemplata inizialmente era quella di piccolo web server, funzione che è poi però risultata meglio implementabile su altre macchine, non necessariamente riservate a Planck/LFI. Di contro altri servizi, come dettagliato più avanti (pag. 6), son stati man mano aggiunti, con l'evidenziarsi nel tempo di specifiche necessità.

[1] LFI è il *Low Frequency Instrument*, strumento progettato e realizzato per lo studio della radiazione cosmica di fondo all'interno della missione ESA Planck [http://sci.esa.int/planck] da un Consorzio Internazionale di Istituti Scientifici a guida italiana (P.I. di LFI è Nazzareno Mandolesi, attuale Direttore dello IASFBO).

2. STRUTTURA DELLA SOTTORETE

Come si può vedere dallo schema generale riportato a seguire (Fig. 1), attualmente il server *max* (A in figura; scheda tecnica in Tav. 1) costituisce, all'atto pratico, il portale d'accesso a tutta la subnet riservata a Planck/LFI. Il che significa anche che, per gli utenti, tutta la subnet stessa è in realtà "trasparente", venendo a identificarsi con il solo *max*.

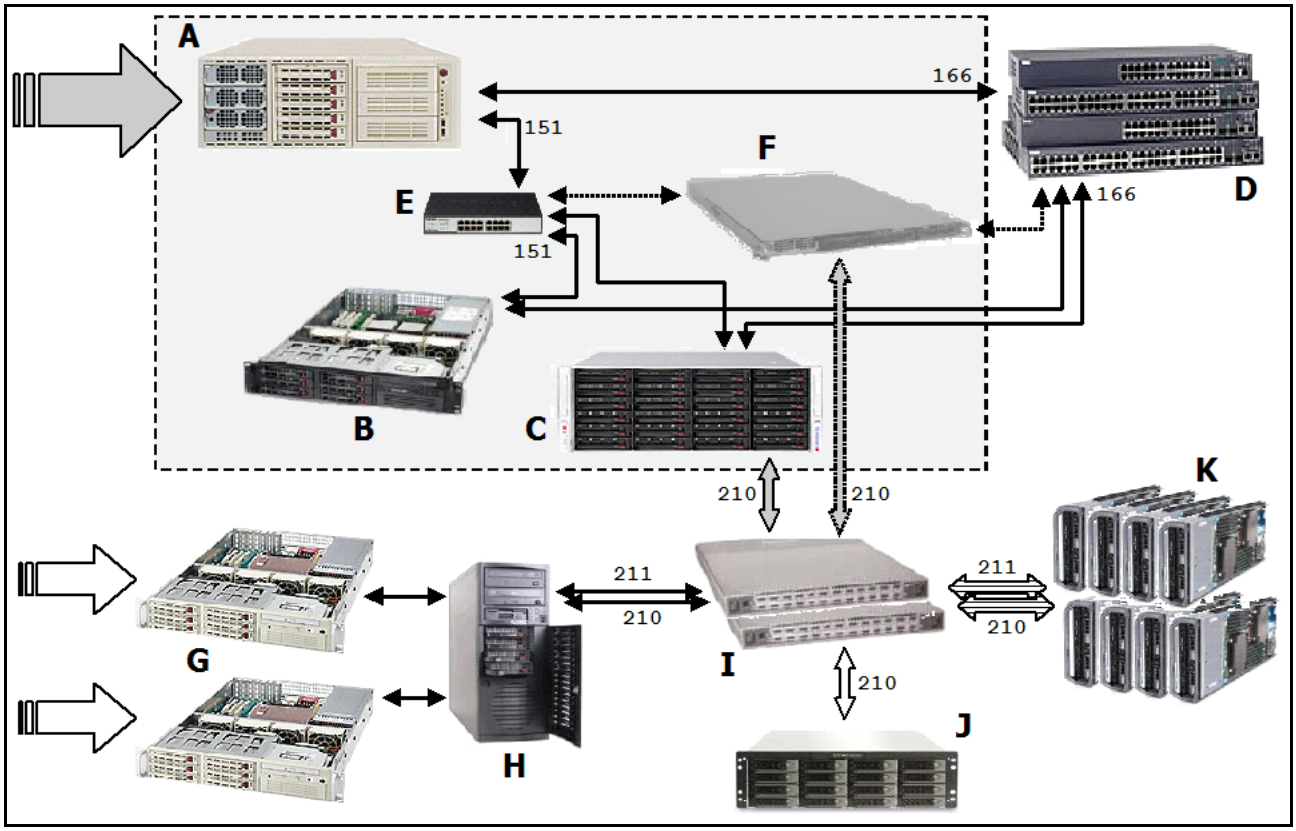

Fig. 1 - Struttura della sottorete e possibili interazioni con l'esterno

```
A = max, primo nucleo e attuale portale di accesso alla subnet
B = karl, primo storage aggiunto (con anche funzioni di backup)
C = ernst, secondo storage aggiunto (con inclusa unità GPFS)
D = switch rack di istituto
E = switch unit per subnet privata
F = server di calcolo dedicato a Planck/LFI [acquisto previsto]
G = generici server di istituto con possibile accesso dall'esterno
H = tonno, server di ingresso al cluster di calcolo d'istituto
I = switch dedicati al cluster di calcolo d'istituto
J = master, server di riferimento per lo storage GPFS
K = nodi del cluster di calcolo d'istituto

151 = subnet 192.168.151.0/24, ad uso esclusivo di Planck/LFI
166 = subnet 192.168.166.0/24, ad uso generico per utenze IASFBO
210 = subnet 192.168.210.0/24, privilegiata per transazioni GPFS
211 = subnet 192.168.211.0/24, riservata ai nodi del cluster
```

Tutta la gestione delle politiche di accesso, a cominciare dalle zone e dalle regole implementate a livello di firewall sino ad arrivare ai criteri di login e all'eventuale *chrooting*, viene in sostanza fatta unicamente su *max*. Il che però non significa, per inciso, che gli altri server della subnet ne manchino completamente: un firewall, ad esempio, è stato comunque configurato su ogni nodo "interno", pur se con regole più lasche, come segue necessariamente dal differente contesto, nettamente meno critico.

Con l'aumentare delle richieste di storage, dopo tre anni, a *max* è stato affiancato un primo server NAS (*Network Attached Storage*), denominato "karl" (B in Fig. 1). E un secondo NAS, ancor più capiente, denominato "ernst" (C in Fig. 1) è risultato urgentemente necessario dopo solo poco più di un anno (in concomitanza col lancio del satellite), con l'evidenziarsi della necessità di avere anche localmente, a Bologna, copia integrale dei dati di missione.

L'interconnessione fra i vari nodi è stata inizialmente basata sulla sola subnet privata disponibile a livello d'Istituto (`*.hide.bo.iasf`, cfr D in Fig. 1). Successivamente però, al drastico aumentare della mole di dati trasferiti attraverso *max*, si è ritenuto opportuno render tale interconnessione più flessibile, aggiungendo un secondo canale basato su di un indirizzamento privato specifico e unicamente riservato alla subnet in questione (cfr E in Fig. 1). Così facendo si è ottenuto di:

- poter meglio bilanciare il carico, anche in funzione dello specifico flusso di dati considerato;
- potersi svincolare, almeno in parte, da possibili condizioni di sovraccarico della subnet d'Istituto;
- ridurre, per quanto possibile, le interferenze con il traffico generico d'Istituto (interferenze altrimenti molteplici e ripetute, anche per un singolo flusso logico di dati);
- aumentare la fault tolerance della subnet: pur se non in modo automatico, un sistema d'interconnessione può comunque in breve tempo, all'occorrenza, sopperire integralmente all'eventuale cedimento dell'altro (si veda in proposito la porzione di `fstab` riportata in Fig. 2).

L'uso di una interconnessione interna alla sola subnet riservata a Planck/LFI ha permesso inoltre di sperimentare con profitto su di essa un *tuning* dei parametri TCP/IP, e in particolare dell'MTU[2], con la conseguente ottimizzazione della banda effettivamente utilizzata[3].

```
# Removable devices here.
/dev/cdroms/cdrom0      /mnt/cdrom      auto            noauto,user             0 0
/dev/fd0                /mnt/floppy     auto            noauto,user,sync        0 0
# NOTE: 'usbfs' definition MUST follow that for 'proc'!!
none                    /proc/bus/usb   usbfs           defaults        0 0
#
# NFS volumes here.
#
karl.hide.bo.iasf:/extra/home/franceschi/pub   /home/franceschi/pub    nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
karl.hide.bo.iasf:/extra/uftp/RAA_repo  /var/uftp/lftp/RAA_FM/repo/     nfs     rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
karl.hide.bo.iasf:/extra/uftp/RCA_repo  /var/uftp/lftp/RCA2xFM/   nfs   rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
#karl.hide.bo.iasf:/extra/uftp/MOC_repo /var/uftp/lftp/MOC_data_/  nfs   rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
ernst.hide.bo.iasf:/extra       /home/derosa/EXTRAs     nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
#ernst.hide.bo.iasf:/extra      /home/franceschi/EXTRAs/raid1   nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
ernst.hide.bo.iasf:/home/hf/avanzo10   /home/franceschi/EXTRAs/lvm-raid6       nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
#ernst.hide.bo.iasf:/Ayrton /disks/Ayrton   nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
#ernst.hide.bo.iasf:/tentera/MOC_repo       /disks/tentera/MOC_repo nfs     rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
#
# the same but using the private NAS servers subnetwork
#
#192.168.151.151:/extra/home/franceschi/pub      /home/franceschi/pub    nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
#192.168.151.151:/extra/uftp/RAA_repo   /var/uftp/lftp/RAA_FM/repo/     nfs     rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
#192.168.151.151:/extra/uftp/RCA_repo   /var/uftp/lftp/RCA2xFM/   nfs   rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
192.168.151.151:/extra/uftp/MOC_repo    /var/uftp/lftp/MOC_data_/ nfs   rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
#192.168.151.152:/extra /home/derosa/EXTRAs     nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
192.168.151.152:/extra  /home/franceschi/EXTRAs/raid1   nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
#192.168.151.152:/home/hf/avanzo10       /home/franceschi/EXTRAs/lvm-raid6       nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
192.168.151.152:/Ayrton /disks/Ayrton   nfs     rw,hard,intr,nfsvers=3,rsize=16384,wsize=16384  0 0
192.168.151.152:/tentera/MOC_repo       /disks/tentera/MOC_repo nfs     rw,hard,intr,nfsvers=3,rsize=8192,wsize=8192    0 0
```

Fig. 2 - Porzione del file `/etc/fstab` relativa ai mount *NTFS*

L'ultimo nodo della sottorete (indicato in Fig. 1 con F) non è in realtà ancora presente: difficoltà di budget ne hanno forzatamente differita l'installazione, che è comunque fortemente auspicabile possa avvenire in tempi, per quanto possibile, brevi. Tale server dovrà fungere da stadio di *preprocessing* specializzato, allo scopo di meglio organizzare il lavoro di gruppo

[2] *Maximum Transmission Unit*, ovvero il numero massimo di byte in un pacchetto IP.

[3] Con un MTU settato a 9000, in logica *"jumbo frame"*, si ha un incremento del *throughput* di poco superiore al 5%, con la quasi saturazione della capacità del canale (prove svolte con `iperf` [http://iperf.sourceforge.net/], tenendo conto delle più svariate – ma comunque plausibili – situazioni al contorno).

nell'elaborazione dei dati scientifici, e di potenzialmente ottimizzare le successive interazioni con il cluster di calcolo d'Istituto[4].

3. *MAX*

Il primo e fondamentale requisito per *max* è stato (e resta tuttora) quello della sicurezza. Per il sistema operativo da adottare la scelta è quindi caduta su di una distribuzione linux relativamente "insolita" (soprattutto qualche anno fa), denominata *gentoo*.
Uno dei principali pregi di questa distribuzione sta difatti nella sua assoluta e totale configurabilità: a fronte di una installazione certamente non banale – soprattutto se confrontata con i procedimenti tipici delle distribuzioni più diffuse –, all'amministratore di un sistema gentoo viene però reso possibile l'effettivo e reale controllo su ogni minimo componente installato. Il sistema operativo viene in sostanza "costruito" pezzo per pezzo, sotto il totale controllo di chi lo installa, e ciò naturalmente permette di alleggerire il sistema di qualsiasi orpello non sia strettamente necessario. Che è poi la miglior premessa per un sistema che debba garantire un buon livello di sicurezza.
Tanto per fare un esempio, su *max* un'interfaccia X è del tutto assente, come anche qualsiasi possibile interfaccia grafica per i tool eventualmente installati.

Per massimizzare la capacità di storage disponibile (cfr Tav. 1), si è scelto di non utilizzare per il sistema operativo i dischi SATA, e di "accontentarsi" per esso di due dischi EIDE, posti quindi poi in una configurazione RAID1 unicamente grazie ad un apposito modulo software, disponibile da tempo nei kernel linux. I dischi SATA han potuto così entrare tutti a formare un'unità RAID5 (questa viceversa gestita a livello hardware) tale da garantire un livello di *fault tolerance* più che buono.
Piuttosto sperimentale è stato anche il criterio con cui si è deciso di organizzare lo spazio di storage. Si è difatti deciso di gestire l'unità RAID5 tramite LVM2[5], scelta che all'epoca si sarebbe potuta forse valutare come un po' azzardata, ma che ha poi dimostrato innumerevoli volte i suoi indiscutibili pregi. Le specifiche su cosa *max* avrebbe dovuto fare, mai del tutto esaustive, e l'indicazione dello specifico servizio che avrebbe dovuto primariamente assolvere, più volte modificatasi col passare del tempo, han reso la scelta dell'approccio LVM2, se inizialmente a malapena "giustificabile", successivamente del tutto motivata. Perché solo grazie all'utilizzo di LVM2 *max* è stato in grado di riconfigurare, di volta in volta, il proprio spazio di storage (senza nemmeno dover esser messo offline!), ed ha potuto così adattarsi in modo del tutto indolore (nonché del tutto trasparente agli utenti) al mutare delle esigenze nel tempo.

```
CPU: Intel Xeon P4 HT 2,40GHz (FSB 533MHz)
M/B: Supermicro P4SSE
chipset: ServerWorks Grand Champion SL
RAM: 1GiB PC266 ECC Registered (2x 512MiB)
System HDs: 2x 80GB EIDE (RAID1/sw)
Storage HDs: 4x 200GB SATA
Controller RAID: 3ware 9500S-4LP (PCI66/64-bit)
Storage CFG: 558 GiB RAID5 (4 HDDs, no hot spare)
FDD: 1,44MB
DVD: DVD+-RW EIDE
NICs: 2x 10/100/1000, 1x 10/100
power supply: 550W, *NOT* redundant
rack size: 19" 4U
linux distribution: gentoo [2005.1 -> 2008.0]
kernel (arch): 2.6.10-hardened-r3 (32 bit)
X-enabled: no
installed on: feb 2005
```

Tav. 1 - Server max: *scheda tecnica*

[4] Si tratterà quindi di un server caratterizzato da una potenza di calcolo piuttosto alta (si è ipotizzata un'architettura a 4 processori quadcore AMD) e da una significativa quantità di RAM (finanche 64 o 128 GB); lo storage, per quanto necessariamente organizzato in modo fault tolerant, potrà viceversa limitarsi a pochi terabyte.

[5] *Logical Volume Manager v2* è la particolare implementazione di gestore di volumi logici usata sui sistemi Linux [http://sourceware.org/lvm2/]; rende la modalità di allocazione di spazio di storage molto più flessibile di quanto possa esserlo con i normali metodi di partizionamento.

Incidentalmente poi, l'unica volta che *max* è dovuto rimanere offline per un periodo apprezzabile (a causa di un aggiornamento con effetti collaterali non adeguatamente valutati, ma forse nemmeno facilmente prevedibili), proprio avere uno storage basato su LVM ha indirettamente consentito (o per lo meno sicuramente accelerato) un recupero totale dei dati – per quanto il procedimento a basso livello che si è dovuto a tal fine seguire sia stato, a dire il vero, tutt'altro che banale.

Come accennato poco sopra, le modifiche nei "*requirement*" per *max*, soprattutto nei primi tempi, son state diverse; e alcune tutt'altro che facili da gestire. Già solo poco tempo dopo l'effettiva entrata in funzione del server, ad es., accanto alla specifica, obbligata e pur sempre valida, di una "garanzia di sicurezza", e in netto contrasto con essa, si è presentata la richiesta (pressante!) di render al contempo possibile un accesso ftp del tutto convenzionale. In altre parole: mentre il requisito sulla sicurezza imponeva, come misura minima, la preventiva registrazione dell'indirizzo IP, veniva richiesto di consentire contemporaneamente e sullo stesso server accessi ftp anche da indirizzi IP viceversa del tutto generici!
Questa sostanziale incompatibilità nei requisiti la si è potuta aggirare solo grazie alle specifiche caratteristiche del demone ftp in uso: `vsftpd`, già a priori giustamente scelto, fra i molti a disposizione, per la sua notevole flessibilità e configurabilità. Un'attenta analisi delle potenzialità di `vsftpd` ha difatti evidenziato la possibilità di usare *"utenti virtuali"*, cioè utenti gestibili in modo non convenzionale, con in particolare password preservate in formato binario e in una posizione configurabile a piacere (in ogni caso diversa da quella degli utenti "ordinari"); una connessione ad un utente reale è per essi comunque inevitabile, ma non deve necessariamente essere univoca, e soprattutto non è percepibile all'esterno. Utilizzando allora utenti virtuali, una porta di accesso ftp non convenzionale (con la definizione di un apposito nuovo "servizio", denominato `uftp`, *"unsecure ftp"*, svincolato dalla canonica porta 21), un accesso rigorosamente *"chrooted"*, e infine aree di storage separate e con privilegi di accesso specifici, si è potuto realizzare quanto richiesto, con giusto un compromesso minimo sul lato della sicurezza intrinseca del server nel suo complesso. I fatti hanno in effetti dimostrato, negli anni a seguire, la validità e l'efficacia di tale soluzione.

```
...
2008-03-07 xinetd.conf revised to allow more connex x sec (for linux net inst)
2008-03-07 ntpd.conf revised; firewall opened for NTP request from "lfi" zone
2008-03-08 added netinst account x insts via FTP (alias for a EF's pub subdir)
2008-03-28 uFTP user rennes2007 become RO; created similar cst-data user as RW
2008-04-28 added sw-guest as uFTP user for occasional needs (typ: sw uploads)
2008-05-14 file systems separately checked and fixed (for RAID1 system volume)
2008-05-14 portage tree resync'd; many packages updated/added (issues raised!)
2008-05-15 many packages emerged, included portage itself: tree resync'd again
2008-05-16 many system packages (gcc 4.1.2, glibc 2.6.1, ....TO BE CONTINUED!)
2008-12-16 SYSTEM FAILURE because of EIDE channels malfunction (or not???)
2008-12-31 SYSTEM recovered, but LVM partitions ISSUE (lvm2 modules updated)
2009-01-06 SYSTEM BACK ONLINE; LVM issue fixed (it was a hard job!)
2009-01-08 NFS client malfunction solved; KARL (NAS) finally fully operative!!
2009-03-04 Firewall rules revisited because of the new range used by DHCP svc
...
```

Fig. 3 - Breve stralcio dal System Log *mantenuto su* max

Molto più spesso i nuovi requisiti imposti in tempi successivi hanno d'altra parte comportato la semplice aggiunta di nuove funzionalità. E' così che *max* è stato utilizzato (per diverso tempo, ora non più), anche come *license manager* in relazione a software con licenza di rete, da render però disponibile solo nell'ambito di Planck/LFI. Ciò, per inciso, a prescinder dall'esistenza di una versione del gestore di licenze, come anche del *vendor daemon*, che fosse specificatamente già certificata per distribuzioni gentoo.
Il server *max* funge poi anche da riferimento per il protocollo di sincronizzazione dell'ora (*Network Time Protocol*, NTP), ma al solito solo per i nodi della subnet Planck/LFI. Tale fun-

zionalità, inizialmente pensata come opzionale e rivolta ai soli utenti, è poi diventata fondamentale per la corretta sincronizzazione di tutti i server NAS della sottorete.
Ovviamente, con l'aggiunta dei NAS, *max* ha assunto anche la funzione di client NFS preferenziale – ove NFS (*Network File System*) è il protocollo usato per esportare i *device* logici sui vari NAS –, con la conseguente necessaria riconfigurazione delle regole del firewall per i servizi implicitamente coinvolti.
Sporadicamente *max* è stato pure utilizzato per sessioni di test nello sviluppo di software per piattaforme linux (come ad es. in occasione della progettazione di moduli di gateway da interfacciare con SCOS2000[6] per il controllo del flusso di dati in ambito I-EGSE[7] di Planck/LFI).
All'occorrenza viene infine anche utilizzato come server di riferimento (grazie alla ricca collezione di immagini ISO di *repository* d'installazione dei più diffusi sistemi operativi col tempo immagazzinate su di esso) per render più efficiente e rapida l'installazione di un nuovo sistema operativo sul generico host del gruppo di lavoro, via rete locale.

Tutte le operazioni di manutenzione e/o aggiornamento di un certo rilievo sono state annotate manualmente in un file di testo apposito, in modo tale da avere un log di riferimento certo al verificarsi di malfunzionamenti o in presenza di anomalie segnalate dall'utenza (un breve stralcio di esempio ne viene dato in Fig. 3).

4. *KARL*

La distribuzione linux installata sul server *karl* (scheda tecnica in Tav. 2) è anch'essa una gentoo, al fine di garantire la massima compatibilità con *max*. Innovativa è però la scelta del dispositivo utilizzato come sede del sistema operativo: una chiavetta USB di soli 2GB.
La scelta è stata indotta dalla necessità di massimizzare la capacità e la fault tolerance dello storage realizzato tramite controller RAID SATA (su *karl* dotato così anche di *hot spare*), a fronte di un budget limitato. La mancanza di fault tolerance a livello di sistema, che sembra viceversa derivarne, viene innanzitutto resa "non critica" grazie al mantenimento di una copia di backup del sistema su di piccolo volume logico appositamente predisposto, e poi, all'atto pratico, quasi irrilevante per il solo fatto di utilizzare un dispositivo che, mancando di parti in movimento, è nettamente meno soggetto a guasti di quanto non lo sia mediamente un disco fisso.
I vari sottosistemi di storage, anche qui gestiti tramite LVM2, vengono resi disponibili all'esterno tramite NFS. Si è a dire il vero valutato anche l'uso del protocollo iSCSI[8], che è però risultato (da un'indagine eseguita all'epoca della scelta,

```
CPU: Intel Xeon QuadCore E5405 2,00GHz (FSB 1333MHz)
M/B: Intel S5000VSA
chipset: Intel S5000V
RAM: 2GiB DDR2 667MHz ECC (2x 1GiB)
System HDs: none (internal USB device)
Storage HDs: 6x 500GB SATA2
Controller RAID: 3ware 9650SE-8LPML (PCI-E x4)
Storage CFG: 1.82 TiB RAID5 (5 HDDs + hot spare)
FDD: 1,44MB (internal USB)
DVD: DVD+-RW DL EIDE
NICs: 2x 10/100/1000
power supply: 600W, redundant
rack size: 19" 2U
linux distribution: gentoo [2008.0]
kernel (arch): 2.6.25-gentoo-r7 (32 bit)
X-enabled: no
installed on: jan 2008
```

Tav. 2 - Server karl: *scheda tecnica*

[6] Il *"Satellite Control and Operation System 2000"* è il sistema configurabile per il controllo missione sviluppato e oramai sistematicamente utilizzato dall'ESA.

[7] I-EGSE sta per *Instrument Electronic Ground Support Equipment*, ovvero l'insieme di hardware e software utilizzato nelle fasi di assemblaggio, integrazione e test degli strumenti progettati per missioni spaziali.

[8] iSCSI, ovvero *Internet Small Computer System Interface*, è il protocollo che estende l'approccio SCSI a dispositivi interconnessi tramite reti TCP/IP.

e con riferimento alla specifica configurazione prevista per il NAS) meno adatto e non ancora così affidabile da farlo preferire comunque, a prescindere da una possibile revisione di altri aspetti già stabiliti. Si è quindi scelto di utilizzare il protocollo NFS, in particolare nella versione 3 (evitando la più recente versione 4, che avrebbe potuto facilmente risultare problematica a causa della non facile aggiornabilità del kernel presente sul server *max*).
Il server NFS è stato opportunamente "condizionato" in modo tale da renderlo compatibile con l'uso del firewall, potendo quest'ultimo venir lasciato "aperto" in modo incondizionato solo sull'interfaccia di interconnessione interna alla subnet, e non certo su quella rivolta verso la sottorete privata d'Istituto (cfr E e D rispettivamente in Fig. 1). Per garantire il buon funzionamento del protocollo NFS, su *karl* è stato infine previsto un meccanismo di polling continuo, opportunamente cadenzato, per la sincronizzazione via NTP con *max*. I "*mount point*" attualmente esportati si possono desumere dalla lista dei dispositivi montati su *max*, riportata in Fig. 4.
Il sistema RAID di *karl*, come già quello di *max*, viene tenuto continuamente monitorato tramite un servizio apposito, che consente anche l'accesso via https (su di una porta specifica) alla console di manutenzione dell'unità RAID. Tale accesso, per quanto soggetto ad un apposito login, viene comunque concesso solo a pochi host ben definiti, in primis a *max* stesso che, essendo fornito di un browser web testuale compatibile con l'interfaccia della console, può all'occorrenza fungere da "ponte". Si è poi predisposto, al verificarsi di situazioni critiche, l'invio automatico di un'apposita mail all'amministratore del sistema, per il cui smistamento è stato dotato di un servizio `postfix` adeguatamente configurato lo stesso *max* (così da poter anche sopperire ad eventuali *failure*, anche solo temporanee, del servizio SMTP offerto a livello d'Istituto).

```
Filesystem            Size  Used Avail Use% Mounted on
/dev/md/2              75G   15G   60G  20% /
/dev/md/0              30M   13M   16M  46% /boot
/dev/mapper/vg0-data  250G   53G  198G  22% /home
/dev/mapper/vg0-xweb   34G  2.5G   32G   8% /var/www
/dev/mapper/vg0-uftp  275G  245G   31G  89% /var/uftp
none                  442M     0  442M   0% /dev/shm
karl.hide.bo.iasf:/extra/uftp/RAA_repo
                      800G  707G   94G  89% /var/uftp/lftp/RAA_FM/repo
karl.hide.bo.iasf:/extra/uftp/RCA_repo
                      800G  707G   94G  89% /var/uftp/lftp/RCA2xFM
192.168.151.151:/extra/uftp/MOC_repo
                      800G  707G   94G  89% /var/uftp/lftp/MOC_data_
karl.hide.bo.iasf:/extra/home/franceschi/pub
                      800G  468G  333G  59% /home/franceschi/pub
ernst.hide.bo.iasf:/extra
                      903G   14G  843G   2% /home/derosa/EXTRAs
ernst.hide.bo.iasf:/home/hf/avanzo10
                      241G  188M  229G   1% /home/franceschi/EXTRAs/lvm-raid6
192.168.151.152:/extra
                      903G   14G  843G   2% /home/franceschi/EXTRAs/raid1
192.168.151.152:/Ayrton
                      3.7T  952G  2.8T  26% /disks/Ayrton
192.168.151.152:/tentera/MOC_repo
                      9.8T  515G  9.3T   6% /disks/tentera/MOC_repo
/home/franceschi/pub/SW/ISOs_et_sim/OSes/SuSE_11'x/openSUSE-11.1-DVD-x86_64.iso
                      4.4G  4.4G     0 100% /var/uftp/temp/franceschi/install
```

Fig. 4 - Una possibile lista di device *montati su* `max`

5. *ERNST*, E LE INTERAZIONI CON ALTRE SUBNET

Il server *ernst* è l'ultimo arrivato ed è chiaramente, all'interno della sottorete considerata, il più potente attualmente disponibile (si veda la scheda tecnica in Tav. 3). Le performance partico-

larmente spinte di *ernst* sono però in realtà anch'esse, forse paradossalmente, il risultato di una carenza di fondi. *Ernst* nasce difatti con la necessità di sopperire all'esigenza di due server diversi: uno che sia semplicemente un ulteriore server NAS asservito a *max*, solo particolarmente capiente (dell'ordine dei 10 terabyte); e un altro che permetta di creare uno storage riservato al gruppo Planck/LFI atto ad interagire al meglio con il cluster di calcolo dell'Istituto.
Da ciò segue l'uso obbligato di una distribuzione diversa dalle precedenti, specificatamente una CentOS, fortemente affine alla Red Hat Enterprise Linux, che è una delle due sole distribuzioni[9] ufficialmente compatibili con GPFS[10], cioè col file system usato per lo storage asservito al cluster di calcolo.

```
CPU: 2x Intel Xeon QuadCore E5520 2,26GHz (QPI 5.86GT/s)
M/B: Supermicro X8DT3-F
chipset: Intel 5520 (ICH10R)
RAM: 6GiB DDR3-1333 ECC Registered (2x 3x 1GiB)
System HDs: 2x 80GB SATA2 (74.53 GiB RAID1)
Controller RAID#0: 3ware 8006-2LP (PCI66/32-bit)
Storage HDs: 24x 1TB SATA2
Controller RAID#1: 3ware 9650SE-12LPML (PCI-E x8) + BBU
Storage CFG#1: 3.64 TiB RAID10 (8 HDDs)
Controller RAID#2: 3ware 9650SE-16LPML (PCI-E x8) + BBU
Storage CFG#2: 931.31 GiB RAID1 (2 HDDs)
Storage CFG#3: 10.00 TiB RAID6 (13 HDDs + hot spare)
FDD: no
DVD: no
NICs: 4x 10/100/1000 (1x 10/100 IPMI-dedicated)
power supply: 1200W, redundant
rack size: 19" 4U
linux distribution: CentOS [5.4]
kernel (arch): 2.6.18-128.7.1.el5.centos.plus (64 bit)
X-enabled: yes
installed on: aug 2009
```

Tav. 3 - Server ernst: *scheda tecnica*

Da ciò segue anche l'uso di due controller RAID distinti: approccio sperimentale, ancora sotto test, finalizzato ad ottenere le performance più adeguate a ciascuno dei due contesti, grazie non solo ai due canali hardware distinti, ma anche alla configurazione fatta a basso livello sui controller, nettamente diversa e quasi diametralmente opposta fra i due, conformemente alle diverse funzionalità richieste ai dispositivi di storage supportati.
Per quel che riguarda il monitoraggio del RAID, l'uso dell'NFSv3 (si veda ancora la Fig. 4), come pure la scelta dell'approccio LVM2, vale né più né meno quanto già detto in precedenza. E' da notare solo come l'uso di LVM2 comporti in questo specifico caso un'ulteriore beneficio: giacché permette, per la sua stessa natura, di sottrarsi alla modalità GPT/EFI (di norma, qualora si vogliano creare volumi più grandi di 2TB, non si può prescindere dall'uso di una *"GUID Partition Table"* e da un'architettura che sia *"Extensible Firmware Interface"*-compatibile) e di mantenere così uniforme la modalità di gestione dei volumi di storage su tutti i server della sottorete.
Il volume di 10 TiB così creato è già da tempo operativo senza aver dato indicazione di alcun problema; archivi di dati di qualche GB ciascuno vengono attualmente aggiunti e controllati in automatico su base giornaliera, provenienti dal MOC[11] tramite il DPC di LFI[12], via canale ftp ad esso riservato.

Un'altra (minima) innovazione posta in atto su *ernst* riguarda l'altro fronte, ovvero l'interfacciamento dell'unità GPFS con il resto dello storage del cluster e con il cluster stesso. Alle due interfacce di rete disponibili a tale scopo è stata cioè applicata una *"link aggregation"* (modalità più frequentemente identificata nello specifico col termine *"bonding"*), con una con-

[9] L'altra è la *SUSE Linux Enterprise Server*.

[10] Il GPFS, *General Parallel File System*, è un file system ad alte performance particolarmente adatto per l'uso con dischi condivisi fra più nodi di calcolo.

[11] Il *Mission Operations Center* sito a Darmstadt (Germania) è l'unità operativa dell'ESA che si fa carico della gestione di tutte le operazioni relative al satellite e agli strumenti posti su di esso.

[12] Il *Data Processing Center* di LFI, sito presso l'Osservatorio Astronomico di Trieste, è responsabile della delivery e dell'archiviazione dei dati scientifici della missione Planck, e in particolar modo di quelli prodotti da LFI.

seguente analoga configurazione dei terminali posti sugli switch del cluster (I in Fig. 1). I test fatti hanno confermato l'efficienza e la validità di tale scelta.
L'interazione con detti switch apre una breccia potenziale nel guscio costruito intorno alla subnet riservata a Planck/LFI. La configurazione del firewall di *ernst* è risultata quindi più delicata di quella del precedente NAS, per quanto in realtà solo leggermente più complessa: come già per *karl* – e contrariamente a quanto fatto per *max*, dove viene usato `shorewall` –, anche per *ernst* l'uso diretto di `iptables` è risultato più che sufficiente. Per la corretta gestione del bonding aperto verso il cluster è bastato difatti aggiungere due sole regole, ad esso connesse, per le due porte riservate all'ssh e al GPFS, rispettivamente. A livello applicativo poi, la comunicazione è stata resa possibile solo a fronte di un opportuno scambio di chiavi asimmetriche fra le parti. Che è poi lo stesso approccio già da tempo adottato per l'accesso ssh su *max* che, dovendo per definizione rimaner possibile da qualsiasi indirizzo IP, è spesso fatto oggetto di *attacchi a dizionario*[13].
E' forse superfluo sottolineare, per concludere, come, a prescindere dal meccanismo utilizzato (le semplici `iptables` o i più complessi script forniti da `shorewall`), a tutti gli accessi cui corrisponde un rischio potenziale segua – quale che sia lo specifico nodo interessato all'interno della subnet – una registrazione univoca in un apposito file di log.

6. CONCLUSIONI

Si può dire che dal momento della loro entrata in servizio tutti i server sopra descritti hanno soddisfatto le aspettative, facendo adeguatamente il loro dovere. Negli ultimi 5 anni si è avuta un'unica interruzione nel servizio (questa non è stata d'altro canto di breve durata; ma che un guasto di una certa entità si sia presentato in prossimità delle festività natalizie non ha certo aiutato!).
L'utilizzo di *ernst* come server "2-in-1" è cosa non ottimale e che continua ad esser sotto osservazione; ma per il momento non sembra dare eccessivi problemi.
La criticità maggiore risiede viceversa, allo stato attuale, proprio in *max*: l'hardware inizia ad esser obiettivamente non più adeguato al carico, che va inesorabilmente aumentando (e che soprattutto ultimamente ha subito un incremento particolarmente significativo); la possibilità di tenere aggiornato un sistema ormai datato, e sempre più tale, si avvia a diventare quasi nulla, diventando sempre più difficile trovare il necessario compromesso fra sicurezza, affidabilità e stabilità; la stessa fault tolerance di *max* presenta infine un tallone d'Achille dato da un'alimentazione non ridondata, che un tempo si poteva ritenere giustamente secondaria, ma che ora, e per entrambi i motivi appena detti, ha assunto la valenza di un fattore di rischio difficilmente ignorabile.
E' per questo che un nuovo server in grado di sostituire *max* (nel modo più indolore possibile) è già in arrivo, nonostante le difficoltà di budget sempre presenti. Si tratta di un'unità 1U ridotta all'essenziale (ma ovviamente con alimentazione ridondata), corredata di un singolo processore di media potenza, e con giusto 4GB di RAM e 2TB di spazio disco complessivo, ottenuto a partire da una configurazione RAID10.
Il nuovo sistema (naturalmente ancora basato su gentoo e privo di interfaccia X) sarà nel suo complesso sicuramente più performante – anche se forse per alcuni aspetti non di molto. E soltanto il sottosistema di storage, pur aumentando in capacità, perderà qualcosa in fault tolerance;

[13] L'*attacco a dizionario* è una tecnica usata per superare un meccanismo di autenticazione (o anche genericamente individuare un codice cifrato), basata su di una serie sistematica di tentativi di inserimento della password, solitamente effettuati in modo automatizzato, a partire da uno o più dizionari contenenti un gran numero di termini potenzialmente validi.

ma ciò lo si può ritenere senz'altro accettabile, e perfettamente in linea con le mutate condizioni al contorno.
Rimane infine la speranza che a breve si possa anche disporre del server di calcolo cui s'è fatto cenno a pag. 4: un ritardo anche solo di un anno, o di anche meno, ne renderà l'acquisto assai meno utile di quanto lo sarebbe adesso.