Public Screens

Luis Miguel Hernández Trigo
E.T. Informàtica Sistemes Juny de 2011

UOC Universitat Oberta de Catalunya

Agraïments a Montse Ariño i a l'equip de debelop.com.

Continguts

1. Presentació i introducció

1.1. Abstract

L'objectiu final és crear un sistema que permeti la visualització d'informació a pantalles públiques. Crearem un sistema amb terminals repartits per diverses ubicacions. Aquests terminals serviran per mostrar informació en determinats establiments comercials segons les necessitats del client.

Els terminals tindran una molt escassa o nul·la interactivitat i seran concebuts per mostrar informació a col·lectius com a les pantalles d'arribades i sortides en una terminal de transports.

La informació a mostrar pot venir d'una font on line pel que necessiten connexió constant.

1.2 Objectius.

Per assolir aquest projecte caldrà dissenyar un node base que s'anirà replicant a les diferents localitzacions i un servidor que pugui monitoritzar i subministrar contingut als nodes remots. La problemàtica de tenir equips en xarxes diverses s'haurà de compensar amb la creació d'una xarxa privada virtual.

1.2.1 Minimitzar les caigudes del sistema

Moltes vegades trobem sistemes que davant de problemes mostren missatges d'error que donen mala imatge a l'usuari final i a l'operador del servei. Uns minuts de downtime o elements absents en la visualització donen la impressió de poca fiabilitat encara que funcioni perfectament el 99% del temps. Per tant s'intentarà en tot moment que els nodes siguin el més robustos possible. Per aquest motiu hem escollit una sèrie d'elements que tenen fiabilitat demostrada. Pel que fa a la xarxa de dades, no podem garantir una disponibilitat del 100% als nodes aïllats, per tant hem optat per dissenyar una solució que permet la desconnexió puntual sense que es faci evident cara al públic.

1.2.2 Riquesa de la plataforma

El sistema ha estat concebut per poder subministrar tot tipus de continguts e informació i ser suficientment versàtil per poder assolir els

requeriments de l’operador del servei. D'aquesta manera podem adaptar-nos a les seves necessitats de disseny així com al tipus d'informació que vol mostrar. Una plataforma web és el més adient per aconseguir-ho.

1.2.3 Facilitat de manteniment

Un sistema informàtic ben administrat sempre comença per una bona monitorització. Tenint eines que permetin generar alertes i un mètode d'accés remot pot donar solució ràpidament a la majoria de situacions de fallada.

1.2.4 Baix cost de desenvolupament

S'ha de fer servir estàndards i llenguatges de programació estesos que permetin trobar desenvolupadors fàcilment. Fer aquest projecte en plataforma web ens facilitarà també aquest punt. Degut a la popularització durant el darrers anys d'aquest tipus de desenvolupaments podem escollir entre un ampli ventall de persones per satisfer les nostres necessitats tant de disseny com de programació a un cost inferior a les altres alternatives.

1.2.5 Actualitzacions de disseny globals

Quan un operador decideix fer canvis en la manera en que mostra la informació a tots els seus nodes s'ha de tenir una eina que permeti

fer-ho de manera global, controlada i simple, sense que es faci evident de cara al públic.

2 Anàlisi

2.1 Elecció del programari

Un servei com el que donarà aquest projecte ha de ser robust, ampliable i alhora ha de tenir el cost el més baix possible. L'ecosistema GNU/Linux ens dóna una sèrie d'opcions que analitzarem per assolir els objectius que ens hem marcat.

2.2 Sistema Operatiu

Òbviament si busquem estabilitat, baix cost per llicencia i una solució àmpliament testada tot ens conduirà a escollir una de les moltes distribucions de GNU/LINUX que existeixen. Ubuntu Linux ens dóna a més a més la possibilitat de manteniment comercial si fos necessari. Als seus repositoris disposa de paquets amb drivers privatius que ens poden ésser útils per donar suport a un determinat hardware. Alguns fabricants posen traves a la pràctica per la creació de drivers opensource. Això és especialment problemàtic quan no podem escollir el hardware a fer servir i més tenint en compte que la part de controladors gràfics és la més afectada per aquest problema. Cal remarcar que encara que hem escollit Ubuntu, aquest sistema s'hagués pogut desenvolupar amb la pràctica totalitat de les distribucions de GNU/Linux amb major o menor dificultat.

2.3 Elecció del sistema gràfic

Existeixen diversos sistemes gràfics per dispositius incrustats. El més conegut és directfb (www.directfb.org). Cada cop es fan servir menys en l'entorn X86 degut a la problemàtica d'adaptar els controladors als nous dissenys hardware gràfics [dfb10], a la flexibilitat guanyada en els últims anys pel projecte xorg i a la superior capacitat del maquinari existent. Per aquest motiu ja ni s'utilitza en els instal·ladors gràfics de les distribucions on tradicionalment dominava. Avui en dia es fa servir xorg en la majoria d'àmbits excepte als smartphones.

La conclusió que en traiem és que podem tenir un entorn gràfic molt més lleuger però que no serà tan madur ni estarà tan testat com xorg i pel qual no tindrem la mateixa oferta d'aplicacions disponibles. Donada aquesta situació optarem per un entorn X11 convencional. Per complementar-lo és convenient disposar d'un gestor de finestres minimalista per fer de contenidor de les aplicacions que hi funcionaran a dins. Tenint en compte aquestes limitades necessitats optem pel gestor de finestres match-box [mat11] que també permet nativament modes a pantalla complerta, doncs va ser dissenyat per dispositius amb pantalles petites. Per aquest projecte només necessitem una única finestra oberta que ocupi tota la pantalla. Descartem així entorns d'escriptori complerts com KDE o gnome.

2.4 Anàlisi de la capa de presentació

Aquesta part és la més critica per nosaltres ja que d'ella dependrà la imatge percebuda pels clients així com els posteriors desenvolupaments a mida per cada operador de servei. Avaluem les diferents possibilitats que ens dóna la comunitat opensource.

2.4.1 Simple DirectMedia Layer

Les opcions a la capa de presentació són múltiples. Sobre X11 tenim a la nostra disposició una ingent quantitat de llibreries, toolkits, widget-sets, etc. Moltes d'elles tenen una sèrie de problemes en comú que les fa no aptes. El problema principal és l'elevat cost de manteniment donada la petita quantitat de representants experimentats en el desenvolupament i el disseny d'aquests entorns dins la comunitat de professionals. Per arribar a aquesta conclusió, analitzem SDL, doncs a priori és la llibreria amb més potencial en aquestes situacions, tal i com diu la seva web [sdl10]:

> "*Simple DirectMedia Layer és una llibreria multiplataforma multimèdia dissenyada per facilitar l'accés de baix nivell a l'àudio, teclat, ratolí, joystick, maquinari 3D mitjançant OpenGL, i framebuffer 2D de vídeo. És utilitzat pel programari de reproducció de MPEG, emuladors, i molts jocs populars...*"

Sembla atractiu i bastant adequat a simple vista. Podríem concebre aquest projecte com si desenvolupéssim un joc, una aplicació gràfica vistosa que funciona a pantalla complerta.

Pros

- Llibertat absoluta de disseny i d'aplicatiu.
- Menor consum de recursos de la màquina.

Contres

- No existeix cap guia per crear disseny. Obligació a crear des de cero.
- Superior complexitat del programa resultant.
- Manca de professionals experimentats en aquest tipus d'aplicatius.
- El desenvolupament, anàlisi d'estabilitat i canvis posteriors són cars i lents.

2.4.2 Sistema de vídeo

Una altra opció és concebre el sistema com un sistema de difusió de vídeo basat en la creació → captura → emissió. El principal avantatge és la simplicitat dels nodes que passarien a ser simples descodificadors de vídeo (mplayer [mpl10], Xine [xin11], dispositius hardware, etc.). Tindria, amb major o menor mesura, els mateixos pros i contres de l'ús de SDL exposats anteriorment, afegint-hi que es necessitarien professionals del sector audiovisual i que es requeriria d'una xarxa potent per permetre la recepció de vídeo [wik10].

Aquesta solució no permet la consecució d'un dels objectius més importants: la tolerància als petits talls de xarxa. Per solucionar això es podria crear un gestor de continguts als nodes que orquestrés els vídeos a emetre i descarregués de nous si pertoca. Això desmuntaria la simplicitat del node que era l'objectiu de fer servir un model basat en difusió de vídeo.

2.4.3 Plataforma web

L'opció plataforma web permet la flexibilitat de desenvolupament en fer servir eines estandarditzades com són PHP i Javascript. En aquest cas funcionant en una arquitectura client-servidor de manera local en un entorn auto-contingut dins una mateixa màquina. Aquest tipus de solu-

ció suposa una reducció de costos degut a la proliferació d'aquest tipus d'instal·lacions i a que permet trobar fàcilment dissenyadors, maquetadors, programadors, etc. amb experiència [sim10]. Les últimes millores en entorns web (HTML5, WebGL[kron11]) permetrien la inclusió de diversos tipus de continguts (vídeo, animacions, etc.) de manera que seria possible crear nous models de disseny per satisfer les necessitats de l'operador del servei.

2.5 Elecció del hardware

En la majoria d'instal·lacions, canviar el cablejat de vídeo o les pròpies pantalles-projectors existents té un cost elevat així que s'oferiran 3 possibilitats diferents a l'operador del servei per intentar minimitzar costos. La primera es composa de node més pantalla, la segona és un node per varies pantalles i l'última només proporciona el programari en un dispositiu USB.

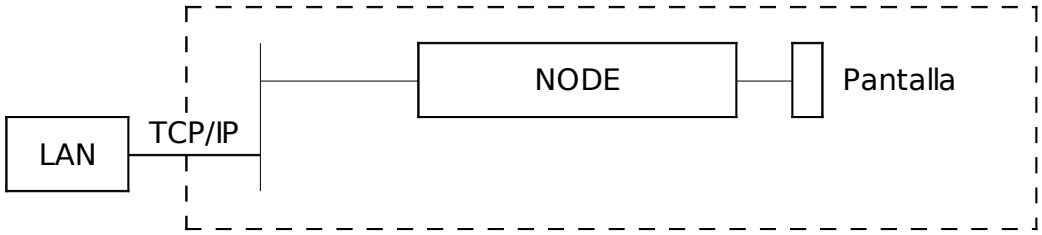
2.5.1 Node amb pantalla

Aquesta primera opció està destinada a petits comerços o àrees molt grans que només es poden cablejar eficientment amb una xarxa de dades en molts casos preexistent. Consistirà en establir un node amb pantalla a cada punt d'informació establint una connexió de dades per

xarxa cablejada o tecnologies sense fils, sense emetre la senyal de vídeo en cap moment. El mateix escenari que tindríem per un únic node.

Exemples:

- Multi-node
    - Aeroport
    - Centre comercial
- Mono-node
    - Farmàcia
    - Ferreteria
    - Parada d’autobús

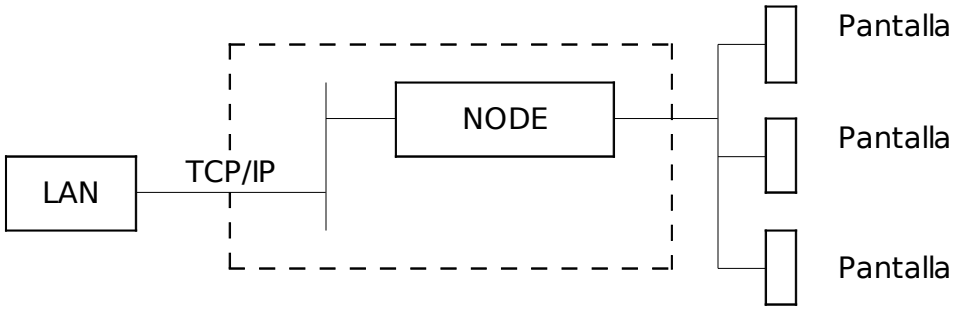


Node amb pantalla

2.5.2 Node únic per instal·lació

Aquest segon escenari parteix de la base que l’operador del servei ja té una infraestructura de cablejat de vídeo al seu establiment utilitzable, en aquest cas s'instal·laria un únic node per a diverses pantalles, convertint la senyal de sortida a S-video, VGA, HDMI o el mètode de con-

nexió existent. Normalment trobarem aquestes instal·lacions en llocs on fan difusió de continguts de vídeo (pel·lícules, clips musicals, tràilers, etc.) i disposen de mètodes de reproducció de vídeo (DVD) com per exemple, museus, discoteques o pubs.



Node amb multiples pantalles

2.5.3 Dispositiu USB

Aquest escenari és el que més aprofita l'entorn ja instal·lat dins l'establiment. En aquest cas trobaríem una instal·lació amb un PC connectat a una o diverses pantalles i només s'hauria d'aportar el software. Com la nostra plataforma està basada en uns components molt concrets que són completament fiables, l'operador del servei rebrà el node remot en forma de pendrive arrencable per fer servir en el seu propi hardware i funcionarà de manera totalment autònoma al programari instal·lat al seu equip ja sigui un sistema basat en Windows o no.

El cost del hardware es òbviament menor en els escenaris on es pot aprofitar més infraestructura però en tenir menys control sobre el

hardware utilitzat s'ha de tenir una especial cura en proveir de tot tipus de controladors que permetin donar servei a la més amplia varietat de hardware possible

És especialment problemàtic el tercer escenari on el sistema arrenca d'un PC amb una VGA connectada a una pantalla, doncs aquestes són totalment desconegudes a priori. És bastant probable que l'engegada inicial sigui suficientment correcta i, si és possible millorar-la, fer-ho.

Per escollir un dispositiu d'emmagatzemament ens basarem en els següents punts a avaluar.

- **Tamany**: Si el nostre dispositiu es petit serà mes fàcil fer-lo arribar a l'operador del servei. També es podrà adaptar millor a l'entorn de maquinari on s'instal·larà.

- **Fragilitat**: Un dispositiu sense peces mòbils que pugui ser enviat per correu ordinari i resisteixi condicions no planificades un cop estigui en funcionament.

- **Compatibilitat**: Cal trobar un sistema que sigui compatible amb el major nombre d'equips existents.

- **Facilitat d'instal·lació**: Encara que la instal·lació serà realitzada per professionals es pot arribar a certs extrems on per un error

greu o altres incidències sigui possible la restitució del servei simplement enviant de nou el dispositiu sencer. Un alt percentatge de la població sabria connectar un USB a un PC.

- **Cost**: El cost dels dispositius USB de memòria flaix amb el tamany necessari es realment baix.

La instal·lació de distribucions de Linux, i de altres SO, és problemàtica sobre dispositius de memòria flaix, l'industria del hardware ens ha ofert aquests dispositius afegint capes d'adaptació per a què puguin ser emprats amb el software actual tot i que a nivell físic són molt diferents dels discos que tradicionalment s'han utilitzat. Tant es així que les optimitzacions que s'han implementat als sistemes de fitxers poden ser fins i tot contraproduents. De moment no existeixen alternatives prou madures per donar format a aquests nous dispositius, tampoc hi ha una alternativa clara per establir una capa inferior (volume manager) que permeti traslladar la fesomia física del dispositiu a la capa superior (la del FS). Aquest problema s'agreuja més amb dispositius USB on tenim una capa addicional que amaga més encara l'estructura física de la memòria flaix [lwn11].

La primera opció és fer servir l'eina usb-creator per crear dispositius arrencables. Aquest sistema no toca ni particions ni dóna format a cap partició sinó que crea un fitxer *.iso* dins del sistema FAT32 del pen-

drive i el combina amb un altre fitxer (loop) que conté un filesystem EXT2 mitjançant AUFS de la mateixa manera que ho faria un live-cd. Aquest mètode no pot funcionar de manera indefinida en el temps ja que, per exemple, no permet actualitzar el kernel del SO ja que tot el que s'instal·la queda a la imatge EXT2 que només és visible un cop Linux ha arrencat i no en el moment de carregar el gestor d'arrancada.

2.6 Elecció de la xarxa

2.6.1 Introducció

No cal remarcar la importància que tenen les VPN en l'entorn de les comunicacions actuals. Les VPN permeten crear xarxes unint punts aïllats fent servir la infraestructura disponible a cada ubicació. Abans de l'aparició de les VPN era necessari establir un enllaç de dades en nodes distants. Això no era percebut com problemàtic degut a que l'abast que tenien les xarxes connectades a Internet era menor. Amb el pas del temps es va estendre aquesta connectivitat pel que era contraproduent i car mantenir les antigues línies dedicades (generalment per PPP) i diversos fabricants van oferir solucions que permetien fer servir la capa 2 del model OSI, la de sessió, com si fos la capa d'enllaç, és a dir, permetien eliminar el cable físic enviant les trames con a dades dins paquets TCP/IP. Els protocols PPTP o L2TP permeten crear aquest enllaços virtuals, anomenats túnels, des de la dècada dels 90. Com les dades ara passaven per nodes fora del control dels usuaris dels túnels, es va aplicar de forma rutinària un xifratge a tots aquests tipus de connexions. Aquest xifratge es considera més segur que una línia dedicada i amb un cost més reduït.

Des del punt de vista de l'administració és molt positiu poder gestionar una xarxa homogènia que tenir equips repartits per diverses ips

publiques d'Internet i podrem relaxar les normes de seguretat dins la xarxa virtual per tal que tots el nodes puguin interactuar més lliurement. Una VPN simple té una topologia d'estrella on tothom es connecta a un concentrador mitjançant túnels.

2.6.2 VPN

Com en aquest projecte ens trobarem amb nodes dispersos per diverses ubicacions ens serà molt útil poder accedir mitjançant una VPN. Així també aconseguirem accedir a qualsevol node remot que estigui darrera un firewall o no disposi de ip pública per trobar-se situat dins una LAN privada. Intentarem trobar una solució dins l'univers de l'opensource que ens permeti crear aquesta xarxa virtual.

De totes les solucions que podem trobar s'han avaluat dues: Openswan i OpenVPN. La primera és una implementació de l'estàndard IPSEC que permet encapsular transmissions dins d'altres per fer túnels i afegeix xifratge i compressió al protocol TCP/IP. La segona es basa en TSL/SSL per proveir un transport segur a les dades que travessaran per l'enllaç virtual.

Tradicionalment s'ha associat IPSEC a les infraestructures TCP/IP i les VPN SSL a la connexió d'usuaris finals a serveis [vpn73]. En el nostre cas és indiferent pensar-ho com una infraestructura d'accés o com acos-

tar serveis d'administració remota als administradors que els han de fer servir.

Les dues estan integrades amb sistemes GNU/LINUX i funcionen creant dispositius de xarxa virtuals que segueixen tots el protocols estàndards d'Internet essent transparents per la resta de components d'un sistema informàtic.

La principal diferencia entre aquestes propostes de la comunitat opensource és la versatilitat. Amb una solució basada en IPSEC es pot arribar a fer infraestructures de xarxa més complexes encaminant paquets des de diferents VPN de manera més homogènia però implica una superior complexitat inicial. Com el nostre cas és molt més simple farem servir la solució basada en TSL/SSL. Cal tenir en compte també que IPSEC es una solució afegida al protocol TCP/IP i encara que forma part dels protocols estàndards d'Internet és possible trobar-nos casos on tinguem problemes degut a que fa servir un tipus de paquets no habituals (ESP, AH) cosa que no succeïx amb OpenVPN on es fa servir només paquets TCP o UDP per tota transacció.

2.6.3 Sistema de monitorització

Nagios és un sistema de monitorització modular que permet establir tests i generar alertes tant de serveis com de problemes a la infraestructura. Té un sistema de plugins ampli i extensible per tenir controlats el serveis habituals així com els propis servidors, switches, routers, fonts d'alimentació i d'altre equipament. Si no trobéssim cap plugin adient per testejar un servei o dispositiu podríem estendre un dels plugins per adaptar-lo a les nostres necessitats.

Un altre component de Nagios és la interfície web, mitjançant un navegador web podem consultar els sondejos configurats per veure si estan donant algun tipus d'error i la gravetat del problema.

Pels problemes greus podrem configurar que els avisos ens arribin per correu electrònic, SMS, etc. També es capaç de mantenir un històric d'alertes per poder investigar problemes que van succeir anteriorment.

El sistema Nagios executa petits scripts anomenats “checks” que verifiquen l'estat dels diferents serveis. Si aquest check ha d'avaluar una condició des de dins de l'equip remot ha d'instal·lar-se prèviament en l'equip i Nagios el cridarà mitjançant una crida remota. Òbviament un servidor ha d'estar escoltant per rebre aquestes crides, es tracta del nrpe-server.

Degut a la simplicitat del projecte, la motorització estàndard que preveu Nagios ens és suficient, no necessitem crear cap plugin nou. No-més caldrà afegir un test per verificar si el port 80 respon. Podrem veure una guia d'instal·lació a l'annex corresponent.

3 Disseny

3.1 Disseny d'arquitectura

Per assolir aquests objectius hem optat per una arquitectura clàssica client-servidor amb la peculiaritat que el client (node) és capaç de funcionar de manera autònoma davant qualsevol desconnexió amb el servidor mitjançant una còpia local de les dades.

ARQUITECTURA

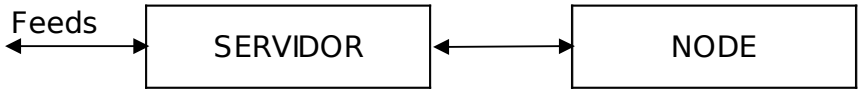
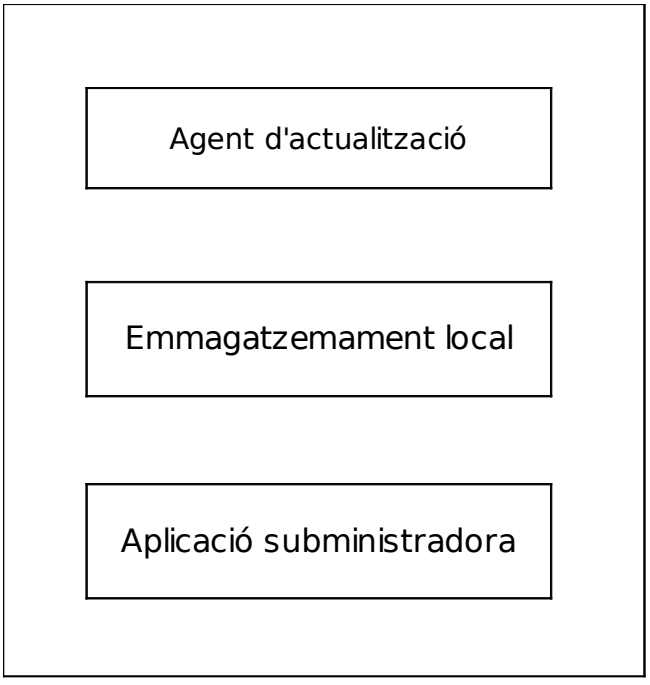


Diagrama de flux.

Tant el servidor com el node estan composats per les mateixes parts:

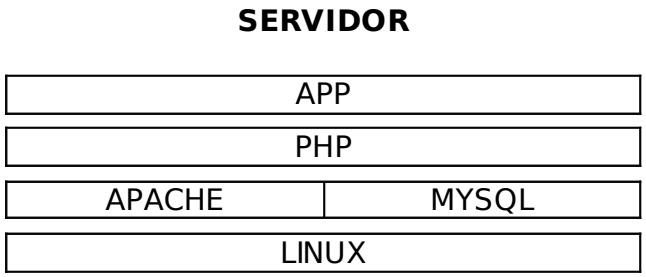


Components

Per conveniència els 3 components es tractaran com una unitat lògica i quedaran allotjats a la mateixa màquina. Si fos necessari, els components del servidor es poden separar molt fàcilment en diferents màquines si la càrrega puja o els aplicatius interfereixen entre ells. A la part dels nodes no es contempla la possibilitat de distribuir la càrrega entre diverses màquines. S'ha d'utilitzar una que permeti donar el servei, doncs qualsevol maquinari existent assolirà les necessitats desitjades àmpliament

3.1.1 Disseny servidor

El servidor, que funciona amb Ubuntu 10.10, servirà les peticions mitjançant una instal·lació d'Apache 2.2 integrat amb PHP emprant mod_php. Aquest PHP ha de tenir suport de mysql que és on recollirà les dades que l'agent de sincronització ha inserit prèviament.



Capes de sofware del servidor

3.1.2 Disseny del client

En l’elecció de l’entorn web del node s'ha optat per establir un servidor web (Apache) integrat amb PHP mitjançant mod_php i amb un backend de BBDD mínim fent servir sqlite3 per la seva maduresa i versatilitat. El dinamisme a la web que mostrarà la informació s'aconseguirà mitjançant Javascript amb diverses llibreries basades en Jquery.

Per mostrar la web del node remot òbviament necessitem un navegador web. La qualitat dels navegadors existents per plataformes GNU/Linux és molt bona i podríem emprar qualsevol d'ells.

S'ha avaluat diverses opcions en les seves diferents versions: Firefox (3.6 i 4.0), Google-Chrome (10.x), Prism, etc. En el nostre cas Firefox 4.0 assoleix amb escreix tots els nostres requeriments. Respecta els estàndards, es ràpid i té un mode a pantalla complerta. Es totalment valid pels nostres propòsits.

NODE

HTML / JS		
WEB BROWSER		
XORG		
LINUX	KMS	DRM

Capes de software del client.

3.1.3 Funcionament lògic del servidor

- L'agent d'actualització sondeja periòdicament la font d'informació carregant les noves dades a la base de dades del servidor.

- Els continguts queden marcats com no llegits a la base de dades amb un identificador que permet trobar-los en funció del node destinatari.

- L'aplicació subministradora rep peticions externes dels nodes amb l'identificador i serveix els missatges amb un token de fi de missatge.

- L'aplicació subministradora proveeix les dades marcant-les com llegides

3.1.4 Funcionament lògic del client

- L'agent d'actualització sondeja el servidor periòdicament subministrant l'identificador corresponent al lloc on està instal·lat.

- Les dades retornades queden guardades com no llegides a la base de dades local.

- L'aplicació HTML/JS crida per AJAX al seu web-server local per demanar més dades.

- L'aplicació PHP retorna les dades i les marca com llegides a la base de dades.

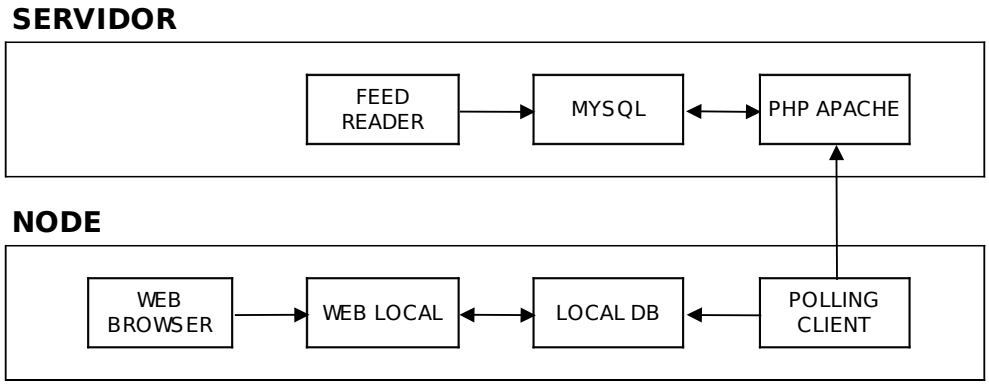


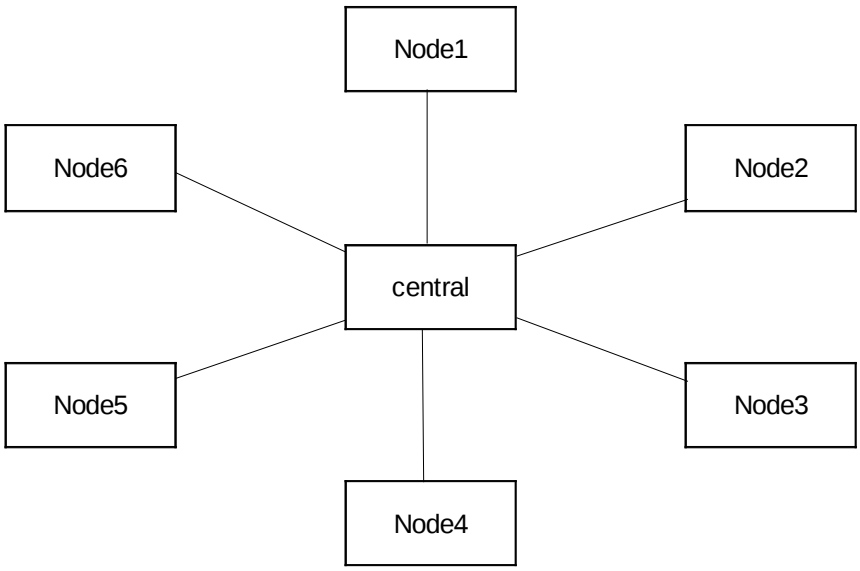
Diagrama lògic.

3.1.5 Topologia de xarxa

Tenim un nombre elevat de nodes repartits per diverses xarxes heterogènies, moltes d'elles sense accés possible des de l'exterior, amb NATs, Firewalls, etc. Ens necessari simplificar aquest escenari cap a una topologia molt més simple.

Escollim la topologia d'estrella amb una única xarxa privada per cada operador de servei.

Aquesta topologia s'aconsegueix mitjançant una xarxa privada virtual openvpn. Totes les connexions es faran molt probablement per enllaços insegurs a traves d'Internet. Diagrama d'estrella amb 6 nodes



Topologia de xarxa.

Per obtenir aquest esquema hem de configurar Openvpn en mode servidor per admetre múltiples connexions dels clients. També activarem la compressió LZO.

De cara al client la única necessitat a assolir es la reconnexió de la VPN si cau l'enllaç físic sobre el qual funciona. Tots aquests detalls de configuració es poden veure al apèndixs corresponents al client i al servidor.

Partirem d'una xarxa privada de classe C 10.8.0.0/24. Openvpn assignarà com a ip del servidor central l'adreça 10.8.0.1 i als nodes ips seqüencials durant la primera connexió, en connexions posteriors intentarà assignar sempre les mateixes ips als mateixos nodes. Aquest mapeig queda reflectit al fitxer ipp.txt i serà la nostra referència per connectarnos als diferents equips.

3.1.6 Requeriments instal·lació del node

Aquesta instal·lació només es farà el primer cop. El següents nodes es crearan clonant el dispositiu sencer sobre un de nou. Cal revisar l'annex de com preparar un llapis de memòria encara que es pot fer durant la instal·lació del SO, és més còmode preparar-ho prèviament.

Podem descarregar Ubuntu 10.10 de qualsevol dels mirrors que tenen repartits arreu del món. No ens interessa la versió per a servidor ja que els nodes són més semblants a un entorn d'escriptori.

Necessitarem apache, php, integrats amb el modul mod-php5 i sqlite com a backend de base de dades pel php. Per que pugui accedir a sqlite s'instal·larà la llibreria php5-sqlite.

La part gràfica ja ve per defecte a Ubuntu així com el navegador. Només requerirem de dues utilitats relacionades: el gestor de finestres matchbox i xwit per amagar el cursor.

Com a utilitats bàsiques de qualssevol sistema en producció tindrem present:

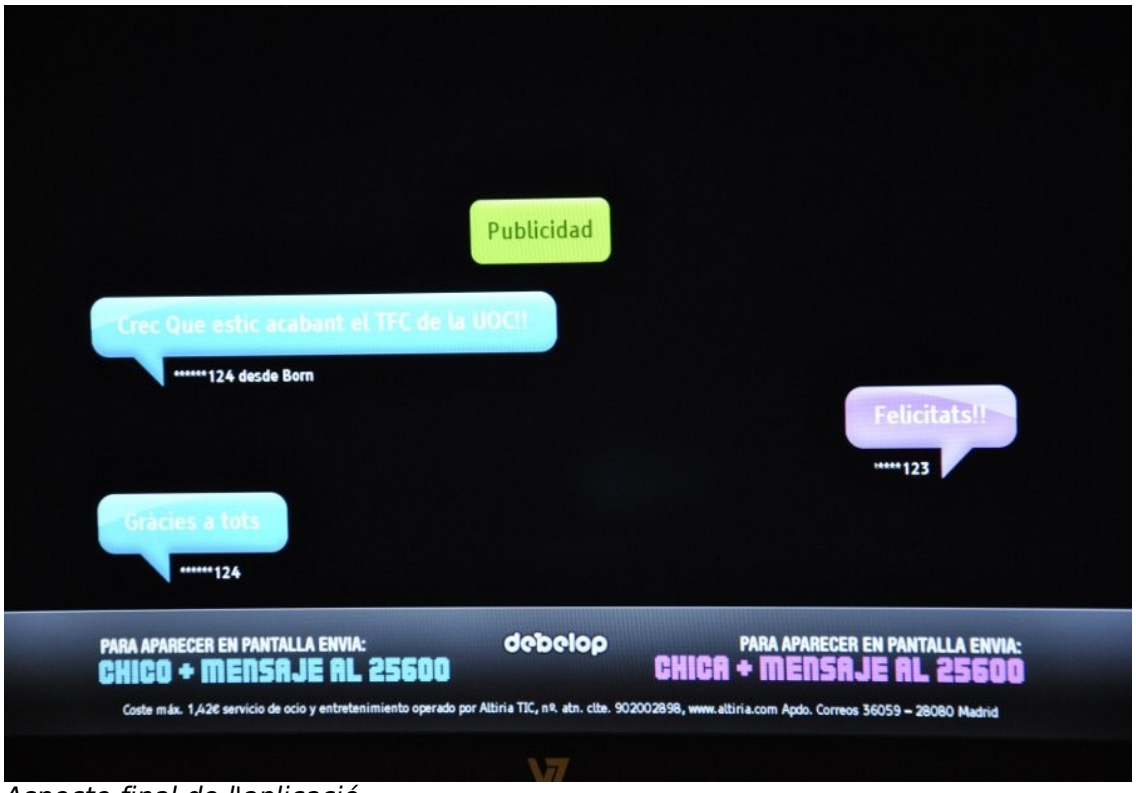
- openssh: el servidor de ssh per poder administrar remotament.

- wget i curl: com a clients http per fer probes i emprar-los dins scripts.
- bsd-mailx: ens aporta la comanda mail per enviar correus des de la línia de comandes
- cvs: per fer actualitzacions de codi
- dd_rescue: per clonar discos.
- mc: com a gestor d'arxius és molt útil.
- vim: com a editor de text.
- nmap: per diagnosticar problemes de xarxa
- strace i ltrace: sempre seran útils per generar traces de l'execució d'aplicatius i fer diagnòstics.

4 Resultats

A les següent il·lustracions podem veure el resultat final. Obtenim un sistema capaç de mostrar informació adaptant-se a qualsevol tipus de pantalla i aprofitant les instal·lacions existents al màxim.

En aquesta captura veiem com missatges que provenen d'un gateway SMS es mostren en un disseny tipus chat. Aquesta es només una de les possibles aplicacions possibles.



Aspecte final de l'aplicació.

A la següent imatge veiem un dels nodes connectat a la pantalla, es tracta d'un mini pc amb CPU atom de baix cost.

Imate d'un node amb pantalla.

Hem assolit els objectius que ens havíem proposat àmpliament. L'anàlisi realitzat sobre la viabilitat d'emprar dispositius USB de baix cost no només ha complert les fites establertes si no que obre tot un ventall de noves possibilitat desconegudes abans d'iniciar aquest projecte.

5 Glossari

Dispositiu flaix: Dispositiu de memòria secundaria no volàtil basat en xips EEPROM.

LAMP: Plataforma web més utilitzada actualment composada d'un servidor Linux, amb apache com a web server, PHP com a llenguatge de programació i Mysql fent de backend de base de dades.

Loop (device): Dispositiu de blocs “virtual” que es troba allotjat dins un fitxer.

Node: Dins aquest projecte és l'encarregat de mostrar la informació del servidor central a les pantalles distribuïdes.

Operador (del servei): Dins aquest document és l'agent que té la necessitat de mostrar dades públicament. Aquesta informació pot ser generada per ell mateix o no.

Pantalla: Dins aquest projecte en referirem al dispositiu on finalment es mostrarà la informació per ser mostrada al public. Pot ser un monitor convencional, una TV, un projector, etc.

Plataforma Web: Parlem de plataforma web per referir-nos a aplicacions que es construeixen en un servidor web i es mostren amb un

navegador i estan construïdes amb toolktis, llenguatges i programari especialitzat com per exemple l'entorn LAMP.

Servidor: És l'encarregat d'obtenir i emmagatzemar la informació per ser distribuïda a cada un dels nodes.

6 Referencies/Bibliografia

[kron11] - WebGL, working group WebGL - OpenGL ES 2.0 for the Web obtingut maig 2011 http://www.khronos.org/webgl/

[xin11] - Xine project authors.
http://www.xine-project.org/

[mpl10] - Mplayer The Movie player.
http://www.mplayerhq.hu/design7/info.html

[sdl10] - Simple directmedia layer project members.
http://www.libsdl.org/

[mat11] - Matchbox project
http://matchbox-project.org/

[wik34] – Article a Wikipedia sobre aquesta sistema
http://en.wikipedia.org/wiki/Skia_Graphics_Engine

[goog24] - Patrick Brady .Anatomy & Physiology of an Android (PDF slides) http://sites.google.com/site/io/anatomy--physiology-of-an-android

[vpn73] - Charlie Hosner (2004) OpenVPN and the SSL VPN Revolution http://www.sans.org/reading_room/whitepapers/vpns/openvpn-ssl-vpn-revolution_1459

[dfb10] – Denis Oliver Kropp directfb Supported hardware http://directfb.org/index.php?path=Support%2FGraphics

[wik10] – Advanced video coding H.264/MPEG-4 Levels http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC#Levels

[sim10] – Phil Simon 2010 The Next Wave of Technologies: Opportunities from Chaos Consultable a google books The costs of free software

[lin11] - Linaro kernel working group Flash card survey. https://wiki.linaro.org/WorkingGroups/Kernel/Projects/FlashCardSurvey

[lwn11].- Arnd Bergmann (2011) Optimizing Linux with cheap flash drives http://lwn.net/Articles/428584/

7 Annexes

7.1 Script d'actualització

Aquest script, un cop col·locat al cron, s'executarà cada minut actualitzant dades cada 15 segons. Per HTTP descarrega un nou fitxer en format csv, verifica que tingui el token de final de fitxer que escriu el servidor per prevenir missatges tallats. I si el troba importa les dades a la base de dades. Amb un .import de sqlite.

```
!/bin/bash

. /etc/pscreens.conf
# pscreens.conf
# TOKEN="####STRING_CONOCIDO####"
# LOGFILE="/datos/pscreens/main.log"

for i in `seq 3`
do

     RFILE=/tmp/newsms.${RANDOM}.tmp
     RFILE2=/tmp/newsms.${RANDOM}.tmp

     wget ${URLBASE}?screen_id=${SCREEN_ID} -O ${RFILE}

     grep ${TOKEN} ${RFILE} && ( grep -v ${TOKEN} ${RFILE} > 
${RFILE2} ; echo -e ".separator \"\|\" \n.import ${RFILE2} 
sms\n" | sqlite3 ${PREFIX_SITE}smsdb.sqlite )

     rm ${RFILE}
     rm ${RFILE2}

     sleep 15
done
```

7.2 Script d'arrencada

Script per arrencar la part gràfica. En ordre, arrenca l'entorn de X-window, el window-manager matchbox, el navegador web i per últim desplaça el cursor per que no sigui visible.

```
#!/bin/bash

case "$1" in
     start)
     export DISPLAY=:0
     /usr/bin/X -ac :0 &
     sleep 3
     /usr/bin/matchbox-window-manager &
     sleep 1
     /usr/bin/firefox http://localhost/?timeout=3000 &
     /usr/bin/xwit -root -warp 10000 10000
     ;;

     stop)
     kill `pidof X`
     killall matchbox-window-manager
     killall firefox
     ;;

     restart)
     $0 stop
     $0 start
     ;;

esac
```

7.3 Instalació d'un node

Al fitxer de configuració d'apt /etc/apt/sources.list descomentem les línies relatives als repositoris d'universe i multiverse per accedir a tot el software d'Ubuntu. I actualitzem la llista de paquets:

```
sudo apt-get update
```

Escriurem en un txt el llistat de paquets a instal·lar i cridem apt-get per portar-ho a terme. Molts del paquets necessaris s'instal·len com a dependència

```
sudo apt-get install `cat instalar.txt`

-------instalar.txt------

apache2.2-bin
apache2.2-common
apache2-mpm-prefork
apache2-utils
bsd-mailx
curl
cvs
dd_rescue
feh
grandr
gstreamer0.10-ffmpeg
gstreamer0.10-fluendo-mp3
gstreamer0.10-plugins-ugly
libapache2-mod-php5
libaprutil1
libaprutil1-dbd-sqlite3
libaprutil1-ldap
matchbox
mc
nmap
nvidia-current
nvidia-settings
```

```
openssh-server
patch
php5-cli
php5-common
php5-sqlite
rar
sqlite3
unp
vim
xwit
---------------
```

Canviar el nom del host als següents fitxers. És important que cada màquina tingui un nom únic com és natural.

```
/etc/hostname
/etc/hosts
/etc/mailname
```

Creem un dispositiu de swap dins el filesystem. Però no el farem servir si el sistema té més de 512M de RAM.

```
dd_rescue /dev/zero /swap -m512M
mkswap /swap
swapon /swap
```

Dins el fitxer /etc/fstab afegim:

```
#/swap    none swap pri=1     0    0
```

És important minimitzar les escriptures a disc especialment si el dispositiu està basat en tecnologia flaix. Afegirem `noatime` a la columna d'opcions dins /etc/fstab. Així evitem que escrigui el timestamp al inode en cada lectura.

```
UUID=XXXXXXXXX /    ext3 noatime,errors=remount-ro     0    1
```

Instal·lem l'aplicació i fem que s'arrenqui i s'apagui correctament durant el boot.

```
sudo mkdir -p /aplicaciones/
tar xvzf bchat-1.0.tar.gz
ln -s bchat-1.0 bchat

sudo ln /aplicaciones/bchat/bin/bchatctl bchatctl
cd /etc/rc3.d
sudo ln -s ../init.d/bchatctl S92bchatctl
cd /etc/rc2.d
sudo ln -s ../init.d/bchatctl S92bchatctl
cd /etc/rc0.d
sudo ln -s ../init.d/bchatctl K08bchatctl
cd /etc/rc6.d
sudo ln -s ../init.d/bchatctl K08bchatctl
```

S'ha d'instal·lar la part html/js de l'aplicació.

```
sudo mkdir -p /sites/debelop/bchat/client/
cd /sites/debelop/bchat/
export CVSROOT=:pserver:lmhernandez@dev.debelop.com:/debelop/
cvs login
cvs co -d client debelop/bchat/client

# cd /sites/debelop/bchat/client
sudo find -type f -exec chmod -v 644 {} \;
sudo chown -Rv www-data:www-data .
sudo   ln   -s   /bchat/apache/sms.conf   /etc/apache2/sites-
enabled/001-sms.conf

# cd /sites/bchat/client
# php install.php
```

És molt important que la base de dades de sqlite pertanyi al mateix usuari en què s'executa apache per que hi pugui escriure. Per homogeneïtat ho estenem a tot el site.

```
# chown www-data:www-data *
```

Copiem la conf a /etc/ i li assignem un SCREEN_ID corresponent a la seva ubicació.

```
cp /aplicaciones/bchat/bin/pscreens.conf /etc/
```

Afegim l'script al cron.

```
# crontab -e
* * * * * /aplicaciones/bchat/bin/psupdater.sh 2>&1>/dev/null
```

7.4 Tria i preparació d'un dispositiu de memòria flaix.

Instal·lar una distribució de Linux sobre determinats dispositius USB es pot considerar fer un us intensiu i no hem de donar per fet que el fabricant hagi tingut en compte aquest tipus de càrreges de treball. Pel que hem d'obtenir dades internes de l'aparell per tal d'avaluar-lo i, si s'escau, donar-li un format més adient.

El grup linaro.org està treballant per avançar en aquest tipus de problemes. Abans de fer la compra del hardware comprovarem si es troba llistat a la seva llista de dispositius analitzats [lin11] per trobar el més adient dels que es troben al mercat.

Si el nostre dispositiu apareix a aquesta pàgina podem extreure els paràmetres necessaris per fer un format correcte. Si no els haurem de trobar nosaltres mateixos amb la utilitat flashbench que ells mateixos han creat.

És tracta d'alinear la partició al tamany de “Allocation unit” i formatejar el FS per que escrigui en trossos no inferiors a “Write Size Unit”.

Per fer això podem fer servir fdisk. Deshabilitarem tots els controls de compatibilitat amb MSDOS amb les comandes p i u

Mostrem la taula de particions en cilindres i en sectors de 512 veiem com el fabricant ha situat la partició al sector 64 o el que es el mateix als 32768 bytes.

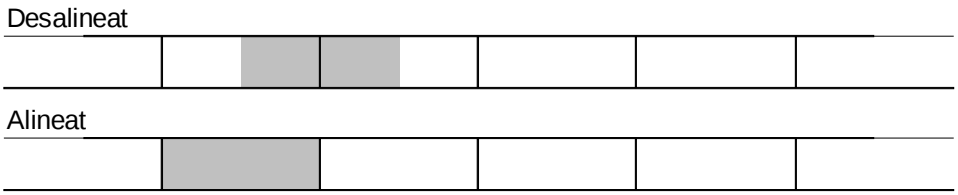
```
$ fdisk /dev/sdb

WARNING: DOS-compatible mode is deprecated strongly recommended to
         switch off mode (command 'c') and change display units to
         sectors (command 'u').

Orden (m para obtener ayuda): p

Disco /dev/sdb: 2023 MB, 2023751680 bytes
21 heads, 20 sectors/track, 9411 cylinders
Units = cilindros of 420 * 512 = 215040 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disposit. Inicio    Comienzo       Fin      Bloques  Id  Sistema
/dev/sdb1               1        9412     1976288    6  FAT16

Orden (m para obtener ayuda): c
El indicador de compatibilidad con DOS no esta establecido

Orden (m para obtener ayuda): u
Se cambian las unidades de visualización/entrada a sectores

Orden (m para obtener ayuda): p

Disco /dev/sdb: 2023 MB, 2023751680 bytes
21 heads, 20 sectors/track, 9411 cylinders, 3952640 sectors en total
Units = sectores of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Disposit. Inicio    Comienzo       Fin      Bloques  Id  Sistema
/dev/sdb1              64     3952639     1976288    6  FAT16
```

Amb la utilitat flashbench trobem que en realitat:

Alocation unit=2Mb

Write Size Unit=128k

Per tant hem d'alinear la partició. Ha de començar en un múltiple de 2Mb. Com estem treballant amb blocs de 512 bytes, s'haurà de col·locar al bloc 4096.



Blocs lògics sobre particions amb diferent inici.

```
# fdisk /dev/sdc

Command (m for help): d
Selected partition 1

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4, default 1): 1
First sector (2048-3952639, default 2048): 4096
Last sector or +size{K,M,G} (4096-3952639, default 3952639):
Using default value 3952639

Command (m for help): p

Disk /dev/sdc: 2023 MB, 2023751680 bytes
21 heads, 20 sectors/track, 9411 cylinders, total 3952640 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1            4096     3952639     1974272   83  Linux
```

```
Command (m for help): w
```

Ara formatem el mitjà informant de que haurà de treballar amb 32 blocs de 4096 cada cop ja que “Write Size Unit” és 128k.

```
mkfs.ext4 -b 4096 -E stride=32,stripe-width=32 /dev/sdc1
```

7.5 Generació de les claus

Afortunadament comptem amb l'eina EasyRSA que ve amb el paquet Openvpn i ens permetrà generar tant certificats com claus publiques i privades.

```
# cd /usr/share/doc/openvpn/examples/easy-rsa/2.0
```

Primer omplim al fitxer “vars” les nostres dades

```
export KEY_COUNTRY="ES"
export KEY_PROVINCE="CAT"
export KEY_CITY="Barcelona"
export KEY_ORG="Debelop"
export KEY_EMAIL="info@debelop.com"
```

Carreguem les variables i esborrem els fitxers que puguin quedar d'execucions anteriors.

```
# . vars
NOTE: If you run ./clean-all, I will be doing a rm -rf on ./keys

# ./clean-all
```

Construïm el fitxer que contindrà els paràmetres Diffie–Hellman

```
# ./build-dh
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.................
```

Com a la nostra VPN no accediran terceres persones no serà necessari fer servir una autoritat certificadora reconeguda. Nosaltres ma-

teixos crearem tot el necessari per poder emetre certificats i claus com si fóssim una autoritat certificadora.

```
# ./pkitool --initca
Using CA Common Name: Debelop CA
Generating a 1024 bit RSA private key
.........++++++
..............++++++
writing new private key to 'ca.key'
-----
```

Creem la key pel servidor central.

```
# ./pkitool --server central
Generating a 1024 bit RSA private key
............++++++
........++++++
writing new private key to 'central.key'
-----
Using   configuration   from   /usr/share/doc/openvpn/examples/easy-
rsa/2.0/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'ES'
stateOrProvinceName   :PRINTABLE:'CAT'
localityName          :PRINTABLE:'Barcelona'
organizationName      :PRINTABLE:'Debelop'
commonName            :PRINTABLE:'central'
emailAddress          :IA5STRING:'info@debelop.com'
Certificate is to be certified until Jun  2 16:25:58 2021 GMT (3650
days)

Write out database with 1 new entries
Data Base Updated
```

Creem la key pel client.

```
# ./pkitool bchat01
Generating a 1024 bit RSA private key
...........++++++
..............++++++
writing new private key to 'bchat01.key'
-----
```

```
Using configuration from /usr/share/doc/openvpn/examples/easy-
rsa/2.0/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'ES'
stateOrProvinceName   :PRINTABLE:'CAT'
localityName          :PRINTABLE:'Barcelona'
organizationName      :PRINTABLE:'Debelop'
commonName            :PRINTABLE:'bchat01'
emailAddress          :IA5STRING:'info@debelop.com'
Certificate is to be certified until Jun  2 16:26:14 2021 GMT (3650
days)

Write out database with 1 new entries
Data Base Updated
```

Els resultats quedaran al directori especificat “keys”.

7.6 Instal·lació de VPN servidor

De les claus i certificats generats en l'apartat anterior necessitarem copiar al directori `/etc/openvpn/` del nostre servidor els arxius següents:

```
dh1024.pem
central.key
central.crt
ca.crt
```

És recomanable fer aquest pas per un canal segur com pot ser SSH. Només restarà preparar la configuració del nostre servidor VPN amb uns paràmetres similars a aquests:

```
--Server.conf--

port 1194
proto udp
dev tun
ca ca.crt
cert central.crt
key central.key
dh dh1024.pem
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
comp-lzo
persist-key
persist-tun
status openvpn-status.log
verb 3
```

I ja podem activar el servei

```
/etc/init.d/openvpn start
```

7.7 Instal·lació VPN del client

De manera similar al client copiarem els següents fitxers:

```
bchat01.crt
bchat01.key
ca.crt
```

Corresponen al certificat del client, la seva clau i el certificat de l'autoritat certificadora.

Configurem el client especificant, el servidor central i sobretot establim que intenti reconnectar indefinidament quan esdevingui una desconnexió.

```
--Client.conf--

client
dev tun
proto udp
remote central.debelop.com 1194
resolv-retry infinite
nobind
persist-key
persist-tun
ca ca.crt
cert bchat01.crt
key bchat01.key
ns-cert-type server
comp-lzo
verb 3
```

Ja podem connectar el servei.

```
/etc/init.d/openvpn start
```

7.8 Instal·lació bàsica de Nagios3

Al servidor instal·lem els paquets corresponents als plugins per les inspeccions locals del servidor, el plugin de nrpe per les probes remotes i la part web.

```
# sudo apt-get install nagios-nrpe-plugin nagios3 nagios-plugins
```

Ens demanarà un usuari i un password per accedir a la part web posteriorment. Podem trobar una configuració d'exemple al propi paquet per integrar-lo amb apache 2, només hem de copiar-la al directori de configuració d'apache.

```
cd /usr/share/doc/nagios3-common/examples/
sudo cp apache2.conf /etc/apache2/sites-available/nagios3
cd /etc/apache2/sites-enabled/
sudo ln -s ../sites-available/nagios3 001-nagios3
```

Dins `/etc/nagios3/conf.d` creem un nou conf per host a monitoritzar. Cada cop que fem modificacions haurem de recarregar Nagios3 (`/etc/init.d/nagios3 reload`).

```
--Node01-chk.cfg--

define host{
      use             generic-host
      host_name       node01
      alias                 Node 01
      address         10.8.0.4
}

define service {
      service_description   HTTP
```

```
    check_command         check_nrpe!check_http
    use                   generic-service
    notification_interval 0
}
```

Al node remot instal·lem els plugins i el servidor que escoltarà les crides nrpe.

```
sudo aptitude install nagios-plugins nagios-nrpe-server
```

Només permetem l'accés des de la ip privada(VPN) del servidor central modificant al fitxer /etc/nagios/nrpe.cfg l'atribut allowed_hosts.

```
allowed_hosts= 10.8.0.1
```

I configurem el nostre test remot dins aquest fitxer.

```
command[check_http]=/usr/lib/nagios/plugins/check_http           -H
http://localhost/
```

7.9 Llicencia

Aquest document esta cobert i es pot distribuir i reproduir segons la llicencia Creative commons CC BY-SA 3.0

http://creativecommons.org/licenses/by-sa/3.0/