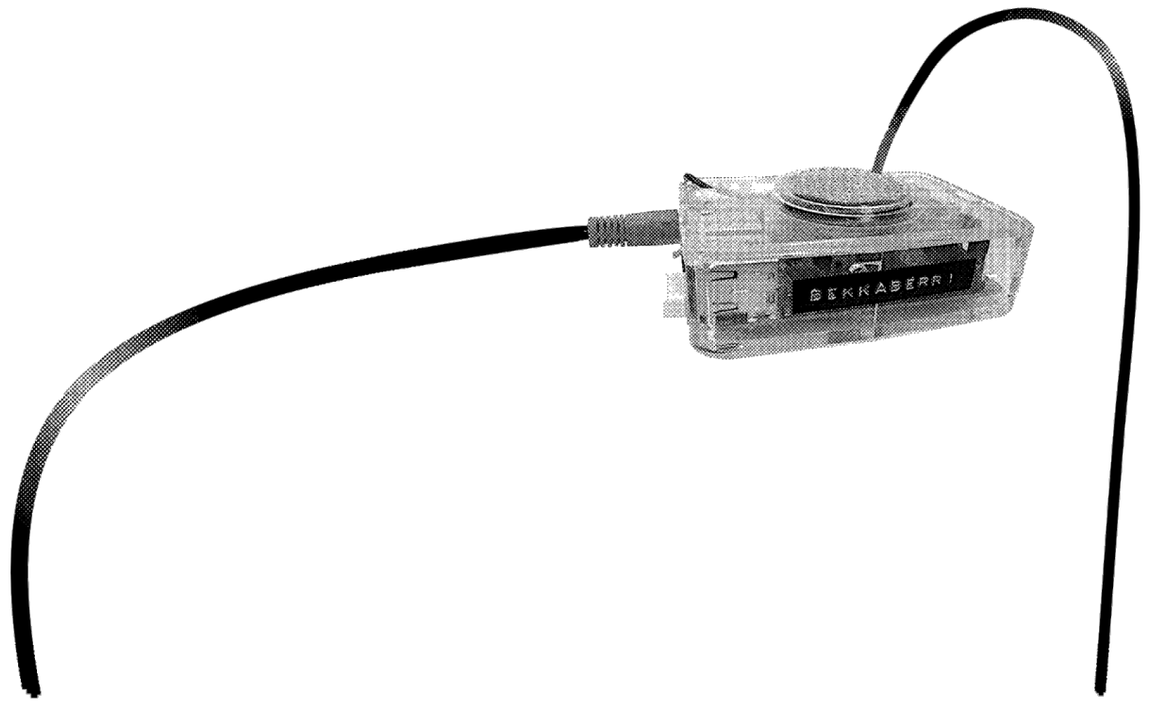
Tot sistema és vulnerable

Una traduccio' (lliure) al català del fanzine de la_bekka per muntar una servidora feminista amb una connexio' casolana.



° la_bekka, octubre 2019
Segona edició: setembre 2023
Traducció catalana desembre 2024

Procés d'aprenentatge: Irene, m4rtu, Inés Binder, Andrea, Marta S.
Redacció i continguts: Inés Binder i m4rtu.
Disseny i maquetació: Inés Binder i m4rtu.
Tipografia: Merriweather, Sunrise International, Open Sans.

Espai per prendre notes, apunta aquí allò que no vulguis oblidar

PRIMERA PART

UNA INFRAESTRUCTURA FEMINISTA

Continguts:

1. Amiguis

Que bé veure'ns per aquí. No sabem com ha arribat aquesta guia a les vostres mans: si us l'ha recomanat algú, si l'heu trobat de casualitat en una fira o si ha aparegut després d'estar navegant d'enllaç en enllaç buscant alguna instrucció més o menys clara sobre com tenir una web pròpia. Sigui com sigui que hàgiu arribat fins aquí, la bona notícia és que ens hem trobat.

Aquesta guia és el resultat de més d'un any de treball a la_bekka, l'espai hackfeminista d'Eskalera Karakola (EKKA), una casa pública transfeminista -antiga casa okupa- de la ciutat de Madrid. Des de finals de l'any 2017 ens ajuntem setmanalment per a aprendre, compartir i muntar la nostra infraestructura digital. Uns mesos més tard, al març de 2018, ens reunim un grup de servidores feministes a Calafou per a conéixer-nos i pensar juntes què significava construir infraestructura feminista. Pensem en les màquines (digitals i analògiques), en codi i en connexions. Però també pensem en nosaltres, en els nostres cossos, les nostres relacions, els nostres temps. Tractem d'identificar què és allò que sosté les nostres

pràctiques: l'aire, l'afecte, l'electricitat, l'empatia, el descans. A dia d'avui encara hi continuem pensant.

De retorn, a Madrid, vam veure que a l'EKKA hi havia equips vells de projectes d'altres anys. Així que vam iniciar un procés d'arqueología informàtica. Revisem tots els equips que trobem, reciclem els que podem i ens en desfem dels altres (portant-los al punt net). Explorem vells discos durs i trobem

Revisem tots els equips que trobem, reciclem els que podem i ens desfem dels altres (portant-losal punt net). Explorem vells discos durs i trobem memòries dels qui havien transitat l'espai abans que nosaltres. Fem còpies de tot, perquè el que s'esborra s'oblida i la memòria alimenta el cor del nostre moviment.

Així que, un cop ens vam enfrontar als equips que teníem, vam començar a somiaen gran. Volíem tenir de tot: des d'un repositori de publicacions i audiovisuals transfeministes fins a instal·lar serveis web que les organitzacions socials poguessin aprofitar amb: pad, calendari, web, llistes de correu, arxius compartits, etc. Ens vam barallar amb les bases de dades, amb el node.js, amb el PHP, amb els rúters GRRR! Finalment, ens vam decidir per començar a muntar una servidoreta que allotgés la nostra pàgina web.

Abans de continuar voldríem aclarir que una servidora no és més que una computadora, normalment connectada a una xarxa o a Internet, que ofereix (acostumem a dir "que serveix") una sèrie de serveis. Si és un servidor web doncs el que serveix és una pàgina web, si és un servidor de correu, el que ofereix és un servidor de correu, etc. Des dels espais transhackfeministes que habitem solem dir "servidora" quan ens referim a una màquina que serveix continguts o serveis feministes. Per això a la nostra maquineta l'anomenem servidora o servidoreta perquè és petita.

La idea d'utilitzar els nostres propis equips, reutilitzant peces velles i cables, aprofitant dispositius barats als quals vam tenir accés, té per a nosaltres un clar fonament polític: **creiem en la sobirania tecnològica, en la nostra capacitat de decidir quina Internet volem habitar.** I la Internet que volem és una Internet transfeminista, ecològica, distribuïda i descentralitzada, al servei dels moviments socials. Una Internet que escapi de les lògiques imposades per Silicon Valley, que ens empeny al consumisme accelerat, la datificació i la mercantilització de cada aspecte de la nostra vida. Volem que la nostra memòria descansi en les nostres màquines, administrades per nosaltres mateixes i sota els nostres principis: l'anonimat, l'aprenentatge i els coneixements col·lectius, la distribució del poder i la cultura lliure. Volem tenir control sobre totes les nostres històries, les nostres receptes, les nostres alegries i tristeses, els nostres mites i llegendes, les nostres cançons i poemes, els nostres imaginaris i fantasies. **Perquè les nostres dades, digitals i analògiques, no poden ser utilitzades per alimentar al monstre.**

Finalment, vam muntar una servidora web utilitzant els recursos que teníem a mà i que vam pensar que eren fàcilment accessibles. Per al maquinari vam escollir una Raspberri Pi, una petita màquina de baix cost a l'abast de moltes persones i amb recursos suficients per a allotjar una web. És cert que no totis tenim la possibilitat de fer-nos amb una. No passa res, podem seguir les instruccions d'aquesta guia utilitzant una computadora vella, una computadora d'escriptori muntada per peces a mode de Frankenstein.

LA NOSTRA MEMÒRIA COL·LECTIVA NO ALIMENTARÀ AL MONSTRE

Una altra dels avantatges d'utilitzar una Raspberry Pi és el seu baix consum energètic. Encara que variï segons els serveis que estigui executant, el consum mitjà per a la nostra servidora serà d'uns 3 watts per hora, el que suposa un consum mensual de 2,15 kilowatts al mes, si la mantenim encesa tot el dia. Així que per més que estigui cara l'electricitat on visquem, no creiem que superi el dòlar/euro mensual.

També utilitzarem Raspberry Pi **OS**, una distribució de **GNU**/Linux desenvolupada per a Raspberri Pi; Apache, perquè és un servidor web amb molta documentació i algunes ja el coneixíem una mica; i Jekyll, un generador de llocs estàtics que redueix la quantitat de recursos necessaris per a funcionar. Aquestes van ser les nostres eleccions a partir del que teníem. Però també poden substituir aquestes opcions per altres, segons el que tinguin a mà o amb el que vulguin experimentar. Poden triar utilitzar Debian o també optar per nGinx per a la seva servidora. Tot és possible, i d'això es tracta: que fem les nostres pròpies eleccions i aprenguem pel camí.

Estem felices d'haver-nos trobat i acompanyar-vos en aquest camí. Esperem que assumiu aquesta tasca amb l'entusiasme amb què ho vam fer nosaltres i aprofiteu aquesta oportunitat per aprendre, construir autonomia i, sobretot, divertir-vos entre amiguis.

Volem agrair a totes les persones que van col·laborar en la construcció d'aquesta guia. A l'**EKKA** per ser la nostra casa, a **C**alafou i a la trobada d'infraestructura feminista per inspirar-nos, a la llista femservers per ser la nostra comunitat de suport, i a Tes, Marta **S.**, **D**ulzet i **S**pideralex per la seva lectura atenta i aportacions afectuoses. Vam ser moltes les que vam dedicar el nostre temps a aquest procés. A totes elles/nosaltres, un reconeixement enorme i agraïment per aquesta petita aportació a la sobirania tecnològica del moviment transhackfeminista.

Una abraçada!

la_bekka, octubre 2019

La porta de l'Eskalera Karakola (Madrid)

2. Instruccions per utilitzar aquesta guia

Escrivim aquestes línies amb la intenció de fer una guia, però que no fos un manual a l'ús. Sovint, les instruccions que trobem en fòrums o blocs donen per fet coneixements que no necessàriament tenim i, a més, la gran majoria estan en anglès. Això genera molta frustració: pensem que res no ens funciona, que no sabem prou o que no servim per això. No us atabaleu, a nosaltres també ens passa! Per això volem explicar amb detall cada pas d'aquest procés, com ens hauria agradat que ens ho expliquessin a nosaltres (i en una llengua coneguda). Per als recursos que només es troben en anglès recomanem el traductor deepl.com que és molt potent i a més respecta la privacitat.

A qui ens adrecem? Si bé és cert que el llenguatge d'aquesta guia és senzill i els passos estan explicats de manera detallada, cal tenir alguns coneixements previs en informàtica. Com a mínim, haver instal·lat prèviament un sistema operatiu. No és per dissuadir a qui no sap res de muntar una servidora web. Al contrari, volem dir-vos que no us preocupeu si les coses no surten bé a la primera, que no us frustreu. Simplement, cal anar més a poc a poc i aprendre pas a pas.

Proposem un recorregut cronològic del procés. Hem pensat en una guia modular, en diferents volums, que ofereixi més agilitat a l'hora de llegir. Vam escollir el format fanzine per la seva versió impresa perquè ens remet a la cultura del **DIY** (fes-ho tu mateixa) i volíem portar l'esperit de l'autoedició al món tecnològic. També tenim una versió digital per poder alimentar aquest text, que esperem que estigui en beta permanent.

Les cinc parts d'aquest fanzine són: 1.Una infraestructura feminista, a mode d'introducció; 2.Hola, maquineta!, per instal·lar el nostre sistema operatiu; 3. De màquina a servidora, per instal·lar Apache; 4.Seguretat per a la nostra servidora, amb mesures i consells de seguretat; 5. La nostra web estàtica amb Jekyll", per aprendre a utilitzar aquest generador de llocs estàtics.

Volem recordar-vos que no som professionals en administració de sistemes. És cert que algunes persones tenim formació en informàtica, però d'altres no.

Hem après durant aquests mesos. Hem posat molt esforç per oferir indicacions que no us portin a situacions de vulnerabilitat. Obrir els ports dels nostres routers pot exposar-nos a atacs si no prenem les precaucions necessàries. Per això hem pres diverses precaucions: hem investigat, preguntat i provat totes les mesures de seguretat que hem pogut per evitar que la nostra servidora ens exposi. Però com aquestes coses canvien constantment, us animem a visitar la nostra pàgina web per llegir qualsevol actualització sobre eines o mesures de seguretat.

Les instruccions d'aquesta guia estan pensades per ser executades des de **GNU/Linux**, concretament **Debian**. Per què? Perquè utilitzem programari lliure i **Debian** és un sistema operatiu lliure que respecta les llibertats de poder utilitzar, distribuir, veure com està fet, modificar i distribuir aquestes modificacions. Moltes d'aquestes instruccions també serveixen per a **MacOS**, ja que també és un sistema basat en **UNIX**. Si utilitzeu **Windows**, oferim alguns enllaços amb instruccions específiques. Això sí, farem servir molt la consola, així que tindrem una excusa per familiaritzar-nos amb l'ús d'aquesta potent eina.

Per editar fitxers de configuració, utilitzarem un editor de text molt senzill anomenat "nano". Hem de recordar tres combinacions de tecles senzilles perquè us sentiu còmodes utilitzant-lo i perquè no sembli que repetim el mateix: per desar, premem **Ctrl + O** (la lletra o). Això mostrarà les línies que estem escrivint i la ubicació on es desaran. Per confirmar, premem **Enter**. I per sortir: **Ctrl + X**.

Us recomanem que documenteu el vostre procés. Aquesta guia és el resultat del nostre, però no hi ha dos processos iguals. Cada procés té els seus propis contextos i les particularitats de les persones que hi participem. Poder documentar les decisions que prenem, els passos que donem i les dificultats que afrontem ens permetrà reconstruir el procés per ajustar-lo, revisar-lo o multiplicar-lo. Ho podeu fer digitalment o de manera analògica amb un quadern.

Finalment, també hem elaborat un glossari en el qual intentem explicar amb paraules senzilles alguns dels conceptes que utilitzarem al llarg de tota la guia. Les paraules que apareguin al glossari estaran ressaltades al llarg del text per tal que pugueu consultar-les. Ningú no neix sabent-ho tot, i ningú no ha de conèixer-ho tot. Per això estem aquí: per formar part d'aquesta construcció col·lectiva.

Qualsevol dubte, aportació o pregunta, podeu adreçar-la al nostre correu hacklabfeminista@riseup.net. També podeu descarregar la nostra clau GPG de https://labekka.red/la_bekka-public.asc. El fingerprint és 242D 8159 9124 E231 EC51 B813 0E4C A504 9F28 BAEB.

3. Què necessitem?

Una Raspberry Pi 3 model B+
(Suposem que es pot fer servir amb altres models. Vam tenir problemes amb una Pi 1 Model B+ a l'hora de configurar Certbot).

Una targeta micoSD per a la Raspberry Pi. Mínim 8GB, recomanat 32GB. Per regla general, aquestes microSD vénen amb un adaptador **SD** per poder-les utilitzar en els nostres ordinadors.

Transformador per connectar la Raspberry Pi a l'electricitat.

Connexió a Internet.

- **Connexió a la xarxa elèctrica.**
- **Accés** d'administradora al **router** pel qual arriba la connexio' a Internet.
- **Cable de xarxa.**
- **Computadora,** només per a la configuracio' i cada vegada que vulgueu actualitzar contingut de la vostra web.
- **Lector de targetes** microSD, pot ser que l'ordinador ja el tingui integrat.
- **Monitor amb sortida HDMI,** només per a la instal·lacio' del sistema operatiu.
- **Cable HDMI,** només per a la instal·lacio' del sistema operatiu.
- **Teclat,** només per a la instal·lacio' del sistema operatiu.
- **Ratolí,** només per a la instal·lacio' del sistema operatiu.

Un domini o una mica de diners per comprar-ne un (opcional, també podem muntar la servidora sense domini propi).

- **TEMPS**
- **PACIÈNCIA**

- Alguns **coneixements de GNU/Linux** i de xarxes (comandaments bàsiques i sense por!).

- Inquietud i **il·lusió** per la sobirania tecnològica.

- **AMIGUIS,** perquè si el procés és compartit, es porta millor.

Quin programari i biblioteques instal·larem al llarg de tota la guia:

- **Raspbian,** el sistema operatiu per a la nostra màquina.

- **SSH** (client i servidor) per poder connectar-nos de manera segura amb la nostra maquineteta.

- Apache2, per instal·lar la servidora web.

- Fail2ban, per augmentar la seguretat de la nostra servidora.

- **UFW**, un tallafoc per a la nostra servidora.

- **Nmap,** per escanejar ports de la nostra servidora.

- **Z**enmap, una interfície gràfica per a nmap.

- **Ddclient,** per gestionar els **DNS** dinàmics (més endavant sabreu què so'n!).

- **Certbot** + biblioteques, per als nostres certificats **SSL**.

- **Ruby** + biblioteques, per a la nostra pàgina web.

- **Build_essential,** també per a la nostra pàgina web.

Imatge de https://takebackthetech.net/

El futur serà

transhackfeminista

4. Glossari

No us espanteu per la terminologia. Anirem pas a pas i, en cas de dubtes, sempre tenim Internet, les amigues i Wikipedia per ajudar-nos. Aquí expliquem alguns conceptes que ens seran útils:

- **Arxiu o fitxer:** És un conjunt de bits identificats per un nom i la ubicació on estan allotjats. És una analogia dels documents analògics. En GNU/Linux es diu "fitxer" i en Windows "arxiu".

- **Certificat SSL/TLS:** És una firma digital que permet xifrar la informació que s'envia entre el servidor i el navegador que accedeix a la nostra web. Ens assegura que la comunicació no pugui ser interceptada ni llegida, i també garanteix la identitat del lloc web amb què es produeix la comunicació.

- **Consola de comandaments:** És un mètode per donar instruccions a un sistema operatiu a través de línies de text. Es coneix també com terminal, línia de comandes o Shell.

- **Daemon o dimoni:** És un tipus de procés informàtic que s'executa en segon pla sense la intervenció directa de la usuària.

- **Direcció IP:** És una adreça que identifica un dispositiu que utilitza el protocol d'Internet. Té la forma de quatre números separats per punts, per exemple: 91.198.174.192.

- **Adreça IP privada/local:** És l'adreça IP que sol començar per 192.168. XXX.XXX i que identifica un dispositiu en una xarxa interna (també coneguda com a privada o local), fora d'Internet. Per exemple, en una casa, oficina o comunitat.

- **Adreça IP pública:** És una adreça IP amb la qual un dispositiu es "connecta" a Internet i és accessible des de fora de la xarxa local.

- Direcció **MAC:** És una adreça única assignada en el moment de la fabricació, a una targeta de xarxa. Es fa servir per identificar de manera exclusiva la targeta o la placa de xarxa.

- **Directori:** És el que en Windows anomenem "carpetes", un contenidor virtual que agrupa una sèrie

- **Directori root d'Apache:** S'anomena així al directori **/var/www/html** on es copien els arxius **HTML, CSS** i el contingut multimèdia que servirà la servidora web. Allà també es guarda l'arxiu **.htaccess.**

- **Domini d'Internet:** És un nom únic que identifica un lloc web a Internet. Traduïx una adreça IP en paraules més fàcils d'identificar i recordar. Per exemple, el domini de 91.198.174.192 és wikipedia.org.

- **DNS (Domain Name Server):** És el sistema que permet associar un nom de domini a una adreça IP.

- **FTP (File Transfer Protocol):** És el protocol de transferència d'arxius.

- **Firewall (Tallafoc):** És un sistema (de maquinari, programari o ambdós) que bloqueja l'accés no autoritzat a una xarxa.

- **Firmware:** És el programari base (com el sistema operatiu però molt lleuger) que ve instal·lat en dispositius com rúters, càmeres, impressores, etc.

- **Host:** És un dispositiu connectat a la xarxa.

- **HTTP (Hiper Text Transfer Protocol):** Protocol que permet la transferència d'informació a través de la web.

- **HTTPS:** Bàsicament és la versió segura de **HTTP**, que utilitza una capa de xifrat **SSL/TLS**.

- **Interfície de xarxa:** És la placa de xarxa, el maquinari que permet connectar un dispositiu a la xarxa.

- **Interfície gràfica d'usuari (GUI, per les seves sigles en anglès):** És un programa que fa servir imatges i elements gràfics per interpretar la informació i les possibles accions que l'usuària pot executar sobre ell.

- **ISP (Proveïdor de Serveis d'Internet):** És l'empresa amb la qual contractes el servei d'Internet per a casa teva.

- **LAN (Local Area Network):** És una xarxa local que connecta dispositius en una àrea limitada. A casa, els dispositius connectats a un mateix rúter formen una **LAN** o xarxa interna.

- **Logs:** Són els registres o historial de totes les accions o esdeveniments que afecten un procés en un sistema informàtic. En cas d'una fallada, són útils com a evidència i per obtenir informació detallada sobre per què ha passat alguna cosa.

- **Paquet:** És un conjunt de dades que pot referir-se a un paquet de dades en el trànsit d'una connexió o a un paquet de programari.

- **Protocol:** És una sèrie d'instruccions que permeten a dos o més sistemes comunicar-se.

- **Port d'Internet:** És un valor que s'utilitza per a cada aplicació que es connecta a un mateix host. Hi ha molts ports per defecte: 80 per a la web, 21 per a FTP, etc.

- **Servei:** També conegut com a dimoni, daemon o programa resident, és un tipus especial de procés informàtic que s'executa en segon pla, en lloc de ser controlat directament per la usuària.

- **Servidor/a:** És un ordinador (maquinari i programari) que "ofereix" un servei: web, correus, streaming, etc. En aquesta guia, anomenem "servidora" a la Raspberry Pi que allotjarà la nostra web.

- Servidor/a web: És un programa informàtic que permet que una pàgina web allotjada en el servidor sigui accessible des d'Internet a través de peticions HTTP que les usuàries fan des dels seus navegadors: escriure una URL, fer clic en un enllaç, veure una foto, etc.

- **Raspberry Pi:** És un ordinador de placa reduïda i de baix cost desenvolupat al Regne Unit per la Fundació Raspberry Pi, amb l'objectiu d'estimular l'ensenyament de la informàtica a les escoles. El seu preu aproximat és de 30 dòlars. No està clara la seva llicència, però permet el seu ús lliure tant en l'àmbit educatiu com personal.

- **Repositoris:** Són com biblioteques, un espai on s'emmagatzemen, organitzen i mantenen una sèrie de recursos; en aquest cas, programes informàtics, llibreries, etc.

- **Root (arrel):** En els sistemes operatius basats en UNIX, és una superusuària que té tots els privilegis possibles.

- **Shell:** Vegeu "Consola de comandaments".

- **Sudo:** És un programa dels sistemes operatius basats en UNIX que permet a les usuàries accedir temporalment i de manera segura als privilegis d'una superusuària (root).

- **TCP (Transmission Control Protocol):** És el protocol bàsic d'Internet que permet les connexions entre xarxes d'ordinadors i s'assegura que els paquets de dades arribin a destí.

- **TTL (Time to Live):** Expressat en segons, indica la vida útil dels dades en un ordinador o una xarxa abans de ser descartats o retornats al seu origen.

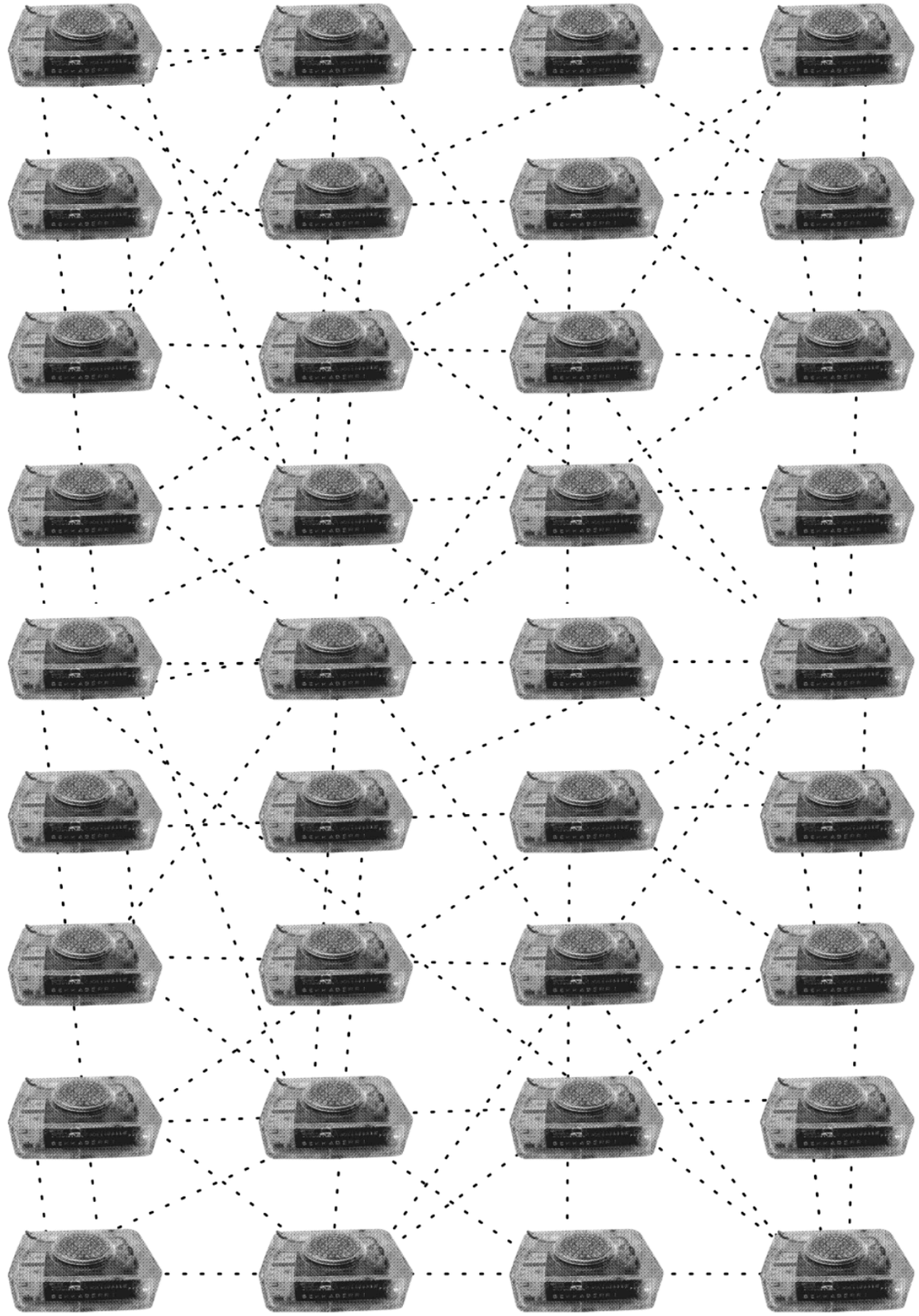
- **UNIX:** És un sistema operatiu que, d'alguna manera, va ser la "mare" d'altres sistemes operatius com GNU/Linux, MacOS, FreeBSD i altres.

- **WLAN (Wireless Local Area Network):** Xarxa interna sense fils, la W inicial prové de "Wireless".

- **WAN (Wide Area Network):** No és exactament el mateix que Internet, però per a aquest tutorial ho podem considerar com a tal.

- **Web:** Atenció! La web no és sinònim d'Internet, sinó només un dels seus serveis basats en el protocol HTTP.

Hi ha algun concepte que no aparegui en aquest glossari? Reviseu la nostra pàgina web https://www.labekka.red o cerqueu a Wikipedia. També teniu unes pàgines a continuació per poder anotar totes les paraules, comandes i trucs que aprengueu al llarg d'aquesta guia. Esperem que la llista s'ompli!



Espai per prendre notes, apunta aquí allò que no vulguis oblidar

Espai per prendre notes, apunta aquí allò que no vulguis oblidar

SEGONA PART

HOLA MAQUINETA!

Continguts:

5. Introducció a **GNU**/Linux
6. Connectem la Raspberry Pi
7. Instal·lem un sistema operatiu a la Raspberry Pi
8. Accedim remotament a la Raspberry Pi utilitzant **SSH**

5. Introducció a GNU/Linux

GNU/Linux, també conegut informalment com a Linux, és un sistema operatiu lliure. El seu desenvolupament és un dels exemples més prominents de programari lliure: tot el seu codi font pot ser utilitzat, modificat i redistribuït lliurement per qualsevol sota els termes de la Llicència Pública General de **GNU** (GPL) i altres llicències lliures.

Hi ha una gran varietat de distribucions de **GNU**/Linux. Potser us sonen noms com **U**buntu, **D**ebian o **M**int. Nosaltres som usuàries de **D**ebian, així que per a aquesta guia hem decidit utilitzar Raspberry Pi **OS**, una distribució per a Raspberry Pi (abans coneguda com a Raspbian) basada en **D**ebian. Per tant, les instruccions d'aquesta guia estan principalment orientades a **GNU**/Linux, una aposta política pel programari lliure i la sobirania tecnològica.

Tot i que per seguir aquesta guia no cal ser expertes, sí que és necessari tenir alguns coneixements bàsics de **GNU**/Linux. Si no els teniu, o són limitats, no us preocupeu. Aquesta pot ser una bona oportunitat per aprendre. Nosaltres hem après molt durant aquest procés!

El procés és més important que el resultat. Gaudiu-ne!

Però repetim: tingueu paciència, consulteu altres fonts i, si teniu dubtes, pregunteu a les vostres amiguis, a la comunitat o a nosaltres.

Repassem alguns comandaments essencials que necessitareu:

pwd: Mostra el directori o ruta on us trobeu.
ls -l: Llista tot allò que hi ha al directori actual i especifica els permisos de cada element.
ls -l / ruta/directori: llista el directori que indiquen i mostra els permisos
ls -la: Llista tot, incloent-hi els fitxers ocults (com **.htaccess**).
cd /ruta/directori: Canvia al directori que indiqueu.
nano /ruta/fitxer: Nano és un editor de textos, així que amb aquest comandament podeu crear i editar un fitxer des de la terminal.
ip a o ifconfig: Mostra les dades de les interfícies de xarxa, inclosa l'adreça IP.

hostname -I: Retorna només l'adreça IP.
chmod XXX /ruta: Canvia els permisos d'un directori o fitxer (lectura, escriptura o execució). Aprendrem més sobre permisos en l'apartat 18.
chown usuària:grup /ruta: Canvia la propietat d'un fitxer o directori assignant una usuària i un grup.
sudo "comando": Permet executar un comandament amb privilegis de superusuària.
apt update: Actualitza la informació dels repositoris.
apt upgrade: Actualitza el sistema operatiu.
apt install paquet: Instal·la el programa que indiqueu.
apt autoremove: Elimina paquets que ja no són necessaris.
service nom_del_servei status/start/stop: Mostra l'estat d'un servei, l'inicia o l'atura.
cp /ruta/origen /ruta/destí: Copia un directori o fitxer d'un lloc a un altre.
mkdir nou_directori: Crea un nou directori.
rm /ruta/fitxer: Esborra un fitxer.
rm -r /ruta/directori: Esborra un directori.
mv /ruta/origen /ruta/destí: Serveix per canviar els noms de fitxers o directoris.
df -h: Mostra l'espai lliure al disc, com la targeta SD de la Raspberry Pi.
grep: és un comandament que ajuda a buscar dins dels resultats d'un altre comandament. Exemple: | grep al final, i a continuació, el que vulgueu buscar

Regla d'or de la terminal: Mai executeu comandaments sense saber què fan al sistema!

Practiqueu aquests comandaments amb el vostre terminal per sentir-vos cada vegada més còmodes. Hi ha molts recursos en línia per aprendre a utilitzar-los. Cerqueu "*Linux commands cheatsheet*" al vostre cercador favorit, i trobareu milers de guies (la majoria en anglès).

És important saber que no totes les distribucions de GNU/Linux tenen els mateixos comandaments. Raspberry Pi OS es basa en Debian, així que tots els comandaments bàsics de Debian funcionaran també en Raspberry Pi OS.

Ara que ja coneixeu els comandaments podeu fer bromes ;) com:
sudo rm -r patriarcat
sudo service machitrolls stop

Hi ha bona documentació sobre comandaments en la web de Raspbe-rry Pi: https://www.raspberrypi.org/documentation/linux/ i https://www.raspberrypi.com/documentation/computers/os.html/, encara que lamentablement només està en anglés.

Recomanem tres recursos per a aprendre a utilitzar la terminal:

Cliteratu: http://www.cliteratu.re/ – Ens introdueix al món de la terminal amb ajuda de la literatura.

Explain Shell: https://explainshell.com/ – Explica detalladament què significa cada part d'un comandament.

The Linux Command Line: Traduït al castellà: https://archive.org/details/la-linea-de-comandos-de-linux

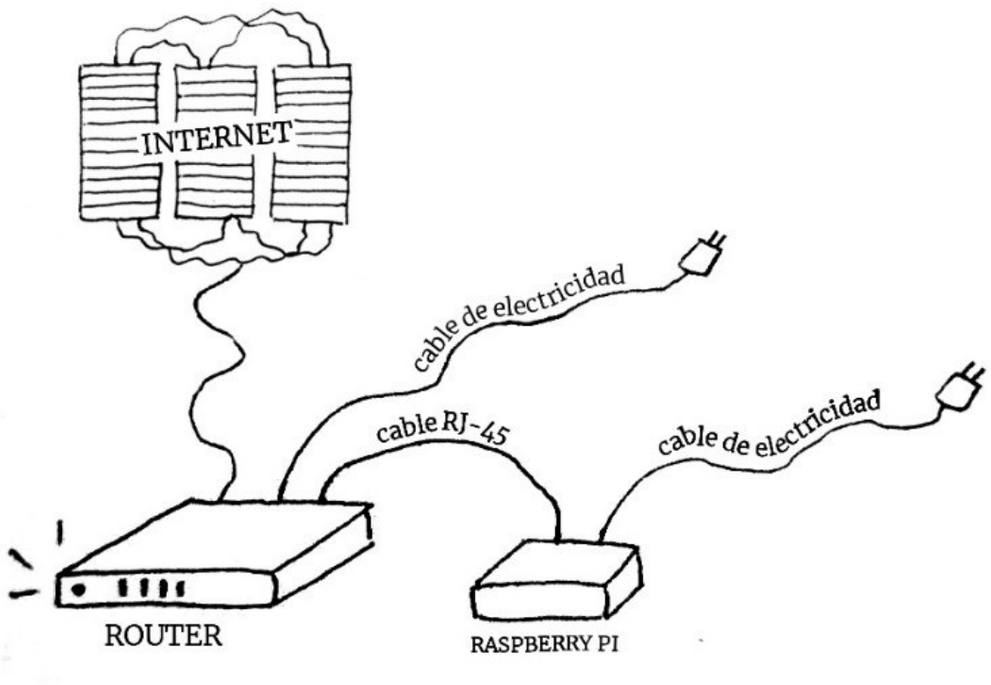
6. Connectem la Raspberry Pi

Abans de començar, creiem que és bona idea començar familiaritzant-vos una mica amb la vostra nova màquina. Sabríeu identificar les diferents parts? Dediqueu una estona a conèixer-la, perquè aquest tros de plàstic i metalls es convertirà en la vostra petita servidora web autònoma.

Si utilitzeu la mateixa Raspberry Pi que nosaltres —la Raspberry Pi 3 Model B+, que en el moment d'escriure aquesta guia era el model més recent—, és important conèixer-ne les característiques bàsiques:

Processador: Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz.
Memòria RAM: 1GB LPDDR2 SDRAM.
Interfície de xarxa sense fils: 2.4GHz i 5GHz IEEE 802.11.b/g/n/ac wireless LAN.
Interfície de xarxa cablejada: Gigabit Ethernet sobre USB 2.0.
Port HDMI: de mida completa.
Ports USB: 4 ports USB 2.0.
Emmagatzematge: depèn de la targeta Micro SD que instal·leu.
Connector d'energia: Entrada de corrent DC de 5V/2.5A.

Ara que ja coneixeu millor la Raspberry Pi, cal decidir on la voleu col·locar. Aquesta decisió és important i cal tenir en compte diverses consideracions, ja que estarà en aquest lloc durant molt de temps.
Connexió i consideracions bàsiques:



Possible esquema de connexió

És imprescindible que estigui connectada al rúter amb un cable d'Internet (RJ-45) i a la xarxa elèctrica, així que no pot estar massa lluny ni del rúter ni d'un endoll. Alguns rúters tenen ports USB, que podrien servir per alimentar la Raspberry Pi directament.

També cal avaluar els riscos i tenir en compte la protecció de la màquina. Com que el lloc web que allotjarem serà accessible des d'Internet, hem de saber que amb l'adreça IP es pot localitzar físicament el lloc on es troba la màquina. Si aquest lloc és casa nostra, això implica un grau de risc. A més, caldrà obrir alguns ports del rúter. Tot i que prendrem mesures per evitar exposar altres dispositius de la mateixa xarxa (com ordinadors o telèfons), sempre existeix un cert risc.

En aquest sentit, és bo valorar el lloc en termes de seguretat física. És recomanable que tinguem accés físic a la màquina, però així com el tenim nosaltres també el poden tenir altres persones:

A l'hora de valorar el lloc físic, considereu:

L'accessibilitat per a altres persones: Que tan fàcil és arribar-hi?

La seguretat del lloc: És estable la connexió de xarxa i elèctrica?

El risc de manipulació accidental: Algú podria desconnectar-la pensant que consumeix molta energia o sense intenció?

Parleu amb la vostra comunitat o col·lectiva i poseu un petit cartell per informar que és important cuidar la màquina. No cal desendollar-la: el seu consum energètic és mínim!

Si disposeu d'un lloc segur, podeu crear el vostre petit altar transhackfeminista i col·locar-hi la servidora connectada al rúter. Nosaltres, per exemple, li vam posar un nom. Feu-la vostra! Els rituals són importants per a nosaltres.

Aquí el nostre petit altar transhackfeminista

7. Instal·lem un sistema operatiu a la Raspberry Pi

El primer que necessitem per comunicar-nos amb la Raspberry Pi són alguns perifèrics: hem de connectar-hi un monitor (o un televisor) mitjançant un cable **HDMI**, un teclat i un ratolí als ports **USB**, i a Internet a través d'un cable de xarxa.

Un cop tinguem el sistema operatiu instal·lat a la nostra Raspberry Pi, podrem accedir-hi de manera remota des d'un altre ordinador. Però, com que encara no tenim res configurat, durant aquest procés d'instal·lació caldrà fer-ho amb la configuració descrita. Després ja podreu deixar la vostra servidora al seu "altar transhackfeminista" sense necessitat de pantalles ni altres cables.
El sistema operatiu que hem decidit instal·lar és Raspberry Pi **OS**, la distribució oficial de **D**ebian per a Raspberry Pi, és a dir, un sistema basat en **GNU**/Linux (si per a la servidora utilitzareu un ordinador convencional, podeu instal·lar **D**ebian directament). Raspberry Pi **OS** es pot instal·lar de diverses maneres, però creiem que fer servir Raspberry Pi Imager és la més senzilla. Només necessitareu connexió a Internet, un ordinador i un lector de targetes **SD**, que molts portàtils actuals ja inclouen. **S**i no en teniu, podeu fer servir un lector extern o, fins i tot, un telèfon mòbil.

Per procedir amb la instal·lació:

1. Inseriu la targeta **SD** de la Raspberry Pi al vostre ordinador.

2. Baixeu Raspberry Pi Imager des de la pàgina oficial de Raspberry Pi **OS**: https://www.raspberrypi.com/software/.

3. Instal·leu-lo al vostre ordinador com faríeu amb qualsevol altre programari.

4. Executeu-lo.

5. **S**eleccioneu el sistema operatiu que voleu instal·lar.

6. A l'opció Raspberry Pi **OS** (other), trieu Raspberry Pi **OS** Lite 64-bit, compatible amb Raspberry Pi 3/4/400.

7. **S**i voleu configurar paràmetres per avançat

7. Si voleu configurar paràmetres per avançat (SSH, WiFi, zona horària, etc.), premeu Ctrl + Shift + X per obrir una finestra de diàleg. Si preferiu configurar aquests paràmetres més tard, simplement podeu finalitzar i seleccionar Write.

   Quan finalitzi el procés:

8. Traieu la targeta SD del lector i inseriu-la a la Raspberry Pi.

9. Enceneu la Raspberry Pi. Tingueu en compte que no té botó d'encesa, sinó que s'encén automàticament quan la connecteu al corrent. Veureu unes llumetes indicatives.

10. Al monitor, veureu que l'instal·lador de Raspberry Pi OS Lite (sense entorn gràfic) comença automàticament. Seguiu els passos indicats durant la instal·lació.

Quan acabi, veureu una pantalla negra semblant al que veiem quan obrim terminal/consola. Apareixerà una línia tipus **pi@raspberry:~$**, que us indica el nom de la usuària i de la màquina.

7.1 Canviem la contrasenya per defecte

Raspberry Pi OS ve amb una usuària per defecte, que és **pi**, i la contrasenya és "raspberry". Com que totes les instal·lacions tenen la mateixa contrasenya, la primera mesura de seguretat que prendrem serà canviar-la.

Escriviu aquest comandament a la consola i premeu *enter*:
sudo passwd pi

Escriviu la nova contrasenya. No us preocupeu si no apareix res mentre escriviu; el sistema ho registra igualment. Confirmeu-la amb *enter.*

Llest! Ara el vostre sistema operatiu té una contrasenya única. Recordeu que una contrasenya robusta hauria de tenir almenys 16 caràcters, incloent-hi majúscules, minúscules, números i caràcters especials. També és important no utilitzar-la en cap altre servei.

7.2 Actualitzem el sistema operatiu

El següent pas serà actualitzar els repositoris. Aquests són bases de dades que contenen programes, llibreries, controladors, etc., que podríem necessitar. Existeixen repositoris oficials, que són els recomanats perquè només inclouen codi verificat i compatible amb el nostre sistema operatiu.

Per actualitzar els repositoris, escriviu:

sudo apt update

Després, actualitzeu el sistema (la Raspberry Pi ha d'estar connectada a Internet):

sudo apt upgrade

Amb això, el nostre sistema operatiu ja estarà instal·lat i actualitzat. Si us sorgeixen problemes o necessiteu més informació, podeu buscar tutorials més detallats a Internet. N'hi ha molts disponibles! :)

En aquest punt, podeu aprofitar per familiaritzar-vos amb l'entorn GNU/Linux. Practiqueu comandaments bàsics, com canviar de directori, crear fitxers de text, llegir-los, editar-los i copiar-los entre directoris. Això us ajudarà a sentir-vos més còmodes amb la terminal.

8. Accedim remotament a la Raspberry Pi amb SSH

Una connexió SSH ens permet connectar-nos a un altre ordinador i treballar-hi de manera remota, com si estiguéssim davant de la màquina, però sense estar-hi físicament. Raspberry Pi OS, des del 2016, no inclou el servei SSH habilitat per defecte, així que l'haurem d'activar manualment. Per fer-ho, hem d'executar el següent comandament a la consola de la Raspberry Pi:

sudo raspi-config

Aquest comandament obrirà una finestra de diàleg que ens permetrà configurar ràpidament algunes opcions del sistema operatiu (contrasenyes, SSH, etc.). Busqueu l'opció *Interfacing Options* i premeu **Enter** per accedir a l'apartat SSH. Us preguntarà si voleu habilitar el servei: *Would you like the SSH server to be enabled?, al que respondrem* "Yes". Després cliquem Acceptar i per sortir del menú anem a *Finish.*

Si necessiteu més informació sobre com habilitar SSH a Raspberry Pi OS, podeu visitar aquesta pàgina:

https://www.raspberrypi.com/documentation/computers/remote-access.html#ssh.

Ara ja podem connectar-nos a la Raspberry Pi per SSH des d'un altre ordinador que estigui connectat a la mateixa xarxa interna, és a dir, al mateix router.

Som-hi! Per connectar-nos per **SSH**, el primer que necessitem és saber l'adreça IP de la nostra Raspberry Pi.

Per connectar-nos per **SSH**, primer necessitem saber l'adreça IP de la nostra Raspberry Pi. Hi ha diverses maneres de trobar-la:

- La manera més senzilla és executar a la terminal de la Raspberry Pi el comandament **hostname -I.** Com a reposta ens retornarà l'adreça IP que necessitem.

- També podeu **trastejar/jugar/experimentar** amb la configuració del rúter per veure quina adreça IP s'ha assignat a la Raspberry Pi.

- Una altra opció és **escanejar la xarxa** amb el comandament **nmap** des d'un altre ordinador connectat a la mateixa xarxa.

- Fins i tot podeu utilitzar una **aplicació per Android** que escanegi la xarxa.

Amb aquestes opcions, segur que trobareu l'adreça IP i estareu més a prop de ser expertes en xarxes!

Un cop tingueu l'adreça IP de la Raspberry Pi, aneu a un altre ordinador, obriu una terminal i executeu el següent comandament:

ssh pi@<adreça-ip>

Per exemple, es veurà una cosa així:

ssh pi@192.168.1.145

Nota: Podeu utilitzar qualsevol sistema operatiu per connectar-vos a la servidora, però recordeu que les instruccions que donem són per a **GNU**/Linux o, en el seu defecte, per a **MacOS**, que també deriva d'**U**nix.

Quan executeu aquest comandament per primera vegada, rebreu una advertència de seguretat per defecte a la que respondrem amb "Yes". Després, ens demanarà la contrasenya de la usuària **pi** (la mateixa que hem assignat anteriorment). Quan la introduïu correctament, estareu connectades a la Raspberry Pi. Igual que si estiguéssim davant d'un terminal al seu monitor. Veureu que la terminal, en comptes de mostrar la usuària i el nom de la màquina que estigueu utilitzant canviarà a **pi@raspberry:~$.** Si voleu saber més sobre com utilitzar connexions **SSH** des de **GNU/Linux** o **MacOS**, podeu visitar: https://www.raspberrypi.com/documentation/computers/remote-access.html#ssh.

8.1 Adéu contrasenyes, benvingudes claus SSH

Ara que ja sabeu com connectar-vos per **SSH** a la vostra Raspberry Pi amb la usuària **pi** i la seva contrasenya, vegem com fer-ho de manera més segura utilitzant **claus SSH**. La diferència és que, en lloc d'obrir la porta de la servidora amb una contrasenya (que qualsevol podria saber), farem servir una clau única per a cada usuària. De manera simplificada, podem dir que pujarem el nostre **cadenat** (clau pública) a la servidora i accedirem amb la nostra **clau** (clau privada). Cada clau és personal i intransferible.

Si no teniu un parell de claus **SSH**, haureu de crear-les. A la computadora que utilitzeu per connectar-vos, executeu el següent comandament: **ssh-keygen -t rsa**

El sistema us demanarà algunes dades, que podeu omplir o deixar en blanc prement *Enter*, i també una contrasenya. Aquesta contrasenya serà necessària cada vegada que us connecteu utilitzant aquest parell de claus. Per defecte, les claus es guardaran al directori **.ssh** dins del vostre directori personal.

- La clau privada es dirà id_rsa.
- La clau pública es dirà id_rsa.pub.

Atenció! Mai compartiu la vostra clau privada. És només per a vosaltres.

El següent pas serà pujar la clau pública (el **cadenat**) a la Raspberry Pi. Això significa que hem de copiar-la al fitxer **~/.ssh/authorized_keys** de la Raspberry Pi. Ho podem fer escrivint aquest comandament a la consola de l'ordinador que utilitzeu per connectar-vos (no a la Raspberry Pi):

ssh-copy-id -i pi@<ip de la raspberry>

Com que encara no hem pujat la nostra clau, ens demanarà la contrasenya de **pi** per establir la connexió. Un cop introduïda i prement **Enter**, la clau pública es copiarà correctament. Provem llavors si podem establir una conexió **SSH**.

Proveu de tornar a connectar-vos, sempre des de la mateixa xarxa local: **ssh pi@<adreça ip>**

Ara, en lloc de demanar-vos la contrasenya de **pi**, us demanarà la contrasenya de la vostra clau **SSH**. Si tot ha anat bé, ja estareu connectades a la vostra Raspberry Pi! Com heu anat veient, aquestes instruccions són per a **GNU/Linux**, però segur que hi ha tutorials que expliquen com generar claus **SSH** des de Windows.

L'últim pas en aquest apartat serà restringir l'accés remot amb contrasenyes. Així, només es podran connectar les persones que hagin pujat prèviament la seva clau **SSH** pública a la Raspberry Pi. Per fer-ho, editem el fitxer de configuració **sshd_config** executant aquest comandament a la Raspberry Pi:

```
sudo nano /etc/ssh/sshd_config
```

Cerqueu la línia on posa **PasswordAuthentication** i canvieu-la a:

```
PasswordAuthentication no
```

Guardeu els canvis amb **Ctrl+O**, premeu **Enter** i sortiu amb **Ctrl+X**. Per acabar, reiniciem el servei **SSH** perquè els canvis siguin efectius executem:

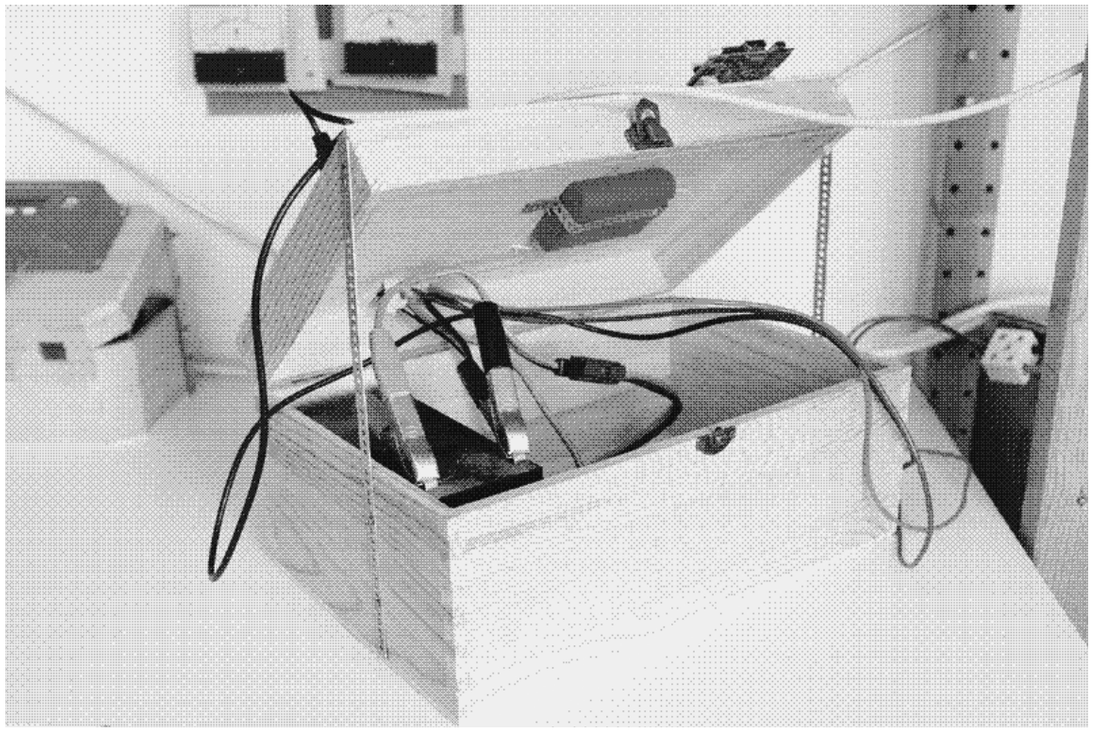
```
sudo service ssh restart
```

Felicitats, amiguis!

En aquest apartat hem aconseguit moltes coses:

- Instal·lar un sistema operatiu a la Raspberry Pi.
- Prendre les primeres mesures de seguretat.
- Habilitar el servei **SSH**.
- Crear les nostres claus **SSH**.
- Connectar-nos de manera remota a la nostra màquina.

Ara ens queda convertir aquesta maquineteta en una servidora!



Primer prototip de la web autoallotjada de Low-tech Magazine amb bateria de plom-àcid (12V 7Ah\) a l'esquerra, i bateria Li-Po UPS (3,7V 6600mA\) a la dreta. *La bateria de plom-àcid proporciona la major part de l'emmagatzematge d'energia, mentre que la bateria Li-Po permet apagar el servidor sense danyar el maquinari.*

TERCERA PART

DE MÀQUINA A SERVIDORA

Continguts:

9. Instal·lem una servidora web a la Raspberry Pi

Hi ha diversos programes per donar vida a una servidora web. Els més coneguts són **Apache** i **Nginx.** Podeu investigar les característiques de cadascun i avaluar quin s'adapta millor a les vostres necessitats. En aquest cas, nosaltres hem triat **Apache** i seguirem els passos per instal·lar-lo.

Bàsicament, consisteix a executar aquest comandament a la terminal de la Raspberry Pi (ja sigui remotament des de l'ordinador o directament des de la Raspberry Pi): **sudo apt install apache2**

sudo apt install apache2

Sí!, així de fàcil. Per comprovar que s'ha instal·lat correctament i que està actiu, podem anar al navegador de l'ordinador i escriure l'adreça IP de la Raspberry Pi (recordeu que encara hem d'estar connectades/des a la mateixa xarxa).

Si no ha canviat, serà la mateixa adreça IP que vau utilitzar per connectar-vos per **SSH**. Si no la recordeu, torneu a executar el comandament **hostname -I** a la consola de la Raspberry Pi.

Hauria d'aparèixer la **pàgina per defecte d'Apache**, que indica que el servidor web està funcionant correctament a la Raspberry Pi. **Emocionant, oi?** Aquesta pàgina que ens mostra s'anomena **index.html**, està guardada al directori "root d'Apache": **/var/www/html/** de la Raspberry Pi. Si voleu substituir-la per la vostra pròpia pàgina web de prova, podeu començar canviant el nom de l'arxiu per fer-ne una còpia de seguretat:

```
sudo mv /var/www/html/index.html /var/www/html/index.html.old
```

Després, podeu crear un nou arxiu **index.html** amb el contingut que vulgueu:

```
sudo nano /var/www/html/index.html
```

Si voleu afegir contingut, necessitareu conèixer una mica la sintaxi d'**HTML**. Podeu començar amb aquest exemple bàsic:

```
<html>

¡Hola amigues!

</html>
```

Guardeu, sortiu i refresqueu la pàgina al navegador per veure si es reflecteixen els canvis. **I voilà!** Si no veieu els canvis, proveu de recarregar-la amb **Ctrl + F5.** Si fins i tot així no es reflecteixen els canvis, proveu d'obrir la pàgina en una pestanya en **mode d'incògnit** o esborreu la **caché** del navegador. De vegades, la versio' antiga de la web queda guardada en memòria **caché.**

De moment, aquesta pàgina web només es pot veure des dels ordinadors que estan connectats al mateix router que la servidora. Encara no és accessible des d'Internet.

En aquest punt, podeu començar a trastejar/experimentar/jugar amb **HTML**. Per exemple, podeu afegir una imatge amb aquest codi: **<img src="http://urldela.imagen.com/foto.jpg">.** Hi ha moooooooooooooooolta documentacio' a Internet sobre **HTML**, plantilles i molts més recursos. Feu proves i divertiu-vos!

Més endavant veurem alguns dels arxius de configuracio' d'Apache, com:

- /etc/apache2/conf-enabled/security.conf
- /etc/apache2/apache2.conf

- /etc/apache2/conf-enabled/security.conf
- /etc/apache2/apache2.conf
- .htaccess

Recordeu: Cada vegada que modifiquem algun paràmetre en un arxiu de configuració, cal reiniciar el servei perquè els canvis es facin efectius. Per reiniciar Apache, executeu:

sudo service apache2 restart

També podeu utilitzar aquest altre comandament:

sudo systemctl restart apache2

Amb això, ja heu donat el primer pas per fer servir la vostra Raspberry Pi com a servidora web.

9.1 Com configurar dues o més webs en una servidora

Ara bé, es pot tenir més d'una web en una mateixa servidora? No només és possible, sinó que no és gaire complicat! Per cada web que vulgueu afegir, necessitareu:

- Un "*virtual host*" o "*vhost*". Es tracta d'un fitxer com aquest: /etc/apache2/sites-available/domini.conf (més avall teniu un exemple).
- Un DocumentRoot o ruta arrel on posar els fitxers que formen cada web. Aquestes rutes solen ser: **/var/www/html/domini1.com/, /var/www/html/domini2.com/, etc.**
  Podeu crear aquesta carpeta amb el següent comandament: sudo mkdir /var/www/html/domini1.com/

Podeu agafar com a exemple de *vhost* el fitxer que ja ve amb Apache: **/etc/apache2/sites-available/ 000-default.conf**

sudo cp /etc/apache2/sites-available/000- default.conf /etc/apache2/sites-available/dominio.conf

A continuació, assegureu-vos d'indicar la ruta a **DocumentRoot** i el domini a **ServerName**. Aquest seria un exemple:

```
<VirtualHost *:80>

    ServerAdmin webmaster@localhost

    ServerName domini.com

    DocumentRoot /var/www/html/domini.com

    ErrorLog ${APACHE_LOG_DIR}/error.log

    CustomLog ${APACHE_LOG_DIR}/access.log

</VirtualHost>
```

Quan tot estigui llest, podeu activar els vhosts amb aquest comandament:

```
sudo a2ensite domini.conf
```

També caldrà desactivar el vhost que Apache té activat per defecte: sudo a2dissite 000-default.conf

Per últim, per aplicar la nova configuració, reinicieu Apache:

```
sudo systemctl restart apache2
```

Si necessiteu més detalls, aquí teniu un tutorial detallat en castellà:

https:// www.digitalocean.com/community/tutorials/como-configurar-vir- tual-hosts-de-apache-en-ubuntu-16-04-es

Nota 1: A la **Cinquena Part** parlarem sobre els permisos que han de tenir els directoris arrel d'Apache, però aquí us fem un avanç:

- L'usuari **pi** ha de formar part del grup **www-data**:

```
sudo usermod -a -G www-data pi
```

L'usuari **pi** i el grup **www-data** han de ser els propietaris del directori arrel:

```
sudo chown -R pi:www-data /var/www/html/domini1.com
```

Els permisos recomanats són **755 per a directoris** i **644 per a fitxers**:

```
sudo find /var/www/html/domini1.com -type d -exec chmod 755 {} \;
```

```
sudo find /var/www/html/domini1.com -type f -exec chmod 644 {} \;
```

Nota 2: Al punt **11** explicarem com configurar Let's Encrypt amb **certbot**. Seguiu els passos i el programa us guiarà en la configuració d'un certificat per a cada domini.

Un cop finalitzat, veureu que **certbot** ha editat el vhost i ha creat un nou fitxer: **/etc/apache2/sites-available/domini-le-ssl.conf**, que indica les noves configuracions necessàries per al certificat.

10. Fem que la servidora sigui accessible des d'Internet

Atenció! Abans de seguir amb aquest apartat, és **imprescindible securitzar la servidora**, ja que en el moment que sigui accessible des d'Internet començarà a rebre intents de connexió. No sempre seran persones intentant accedir-hi, sinó bots automàtics que intenten aprofitar infraestructures alienes. **Recomanem passar directament a la Quarta Part i securitzar la servidora abans de continuar.**

Quan ja tingueu la vostra petita servidora funcionant i securitzada, aprendrem a fer que qualsevol persona a Internet pugui accedir a la pàgina web allotjada a la vostra servidora casolana, situada "darrere" d'un rúter domèstic. És **essencial** repassar tota la **Quarta Part** abans de començar aquesta part, ja que quan surti a Internet estarà vulnerable exposada a possibles atacs.

En aquest apartat aprendrem com configurar el rúter perquè les connexions externes (les que arriben des d'Internet) s'encaminin cap a la servidora i mai cap a un altre dispositiu de la xarxa local. També veurem com associar un domini a la servidora.

El que ve a continuació pot semblar més complex, amb conceptes nous sobre xarxes i protocols. **Poseu-vos còmodes, perquè quan tot funcioni, serà molt emocionant!**

10.1 El rúter de casa o de la comunitat

Comencem accedint al panell de control del router. Recordeu com fer-ho? Heu d'obrir un navegador i escriu-re a la barra d'adreces la IP local del rúter (normalment **192.168.1.1** o **192.168.0.1**). Inicieu sessió amb la usuària i contrasenya configurades en la Quarta Part.

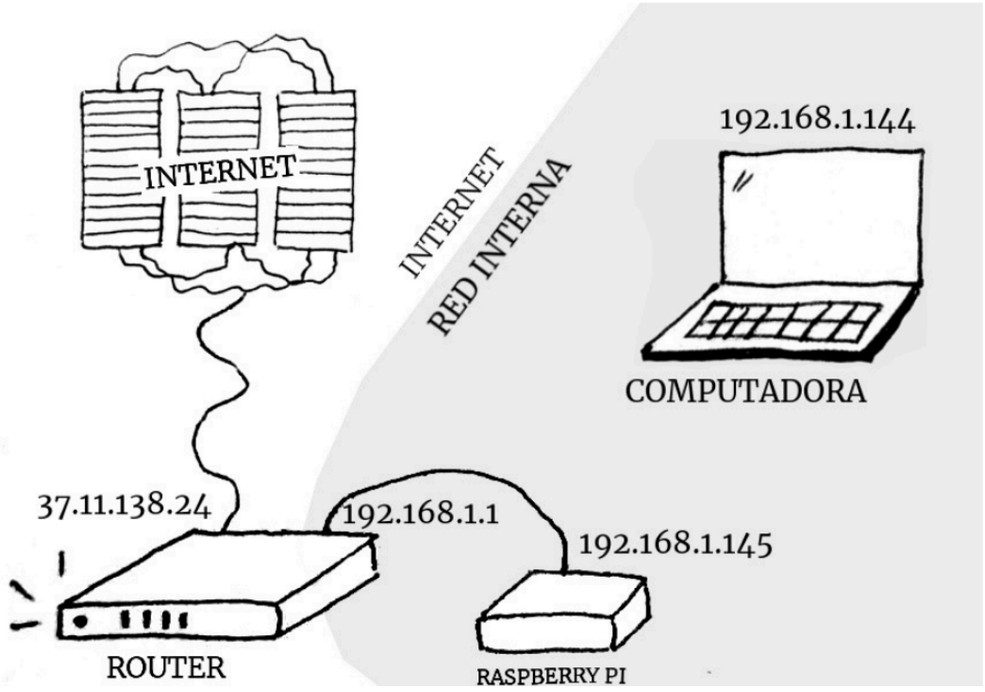
Un cop dins del panell de control del vostre rúter, feu una ullada als diferents menús i opcions disponibles, ara de manera més detallada. Com ja sabeu, aquests menús varien segons el fabricant, així que aquí no podrem guiar-vos pas a pas. **Confieu en la vostra intuïció, un xic d'anglès i alguns manuals d'Internet.** La majoria de routers domèstics són bastant similars.

Un aspecte important dels rúters domèstics és que, a diferència dels mòbils o ordinadors, tenen dues adreces IP:

1. **Adreça IP privada o local:** Identifica el router dins de la xarxa local.
2. **Adreça IP pública:** Identifica el router a Internet.

El que fa el rúter és gestionar i redirigir el trànsit entre la xarxa local i Internet.

- **Trànsit sortint:** Permet que els dispositius de la xarxa local enviïn dades a Internet.
- **Trànsit entrant:** Només permet l'entrada de dades sol·licitades pels dispositius locals.



Intenteu dibuixar un petit esquema que reflecteixi la vostra xarxa local amb les adreces IP de cada dispositiu connectat. A la configuració del rúter, normalment hi ha un apartat on es llisten les adreces IP locals dels dispositius connectats i la IP pública del rúter. Podeu consultar la IP pública des de qualsevol dispositiu connectat al router: Obriu un navegador i cerqueu "Quina és la meva adreça IP?". Entre els resultats, trobareu moltes webs que us diran la vostra IP pública. Nosaltres recomanem https://wtfismyip.com/ o el mateix DuckDuckGo.

Tots els dispositius connectats al mateix router comparteixen la mateixa adreça IP pública.

Si la IP pública que mostra el navegador és diferent de la que veieu al rúter, és possible que el vostre proveïdor d'Internet estigui utilitzant **CGNAT** (Carrier-Grade NAT) per optimitzar el nombre d'adreces IP que assigna entre els clients. Això ens va passar i vam haver de trucar a la companyia i demanar que ens assignessin una **IP pública única** (encara que no fixa) i va funcionar.

Nota: Si la vostra companyia d'Internet utilitza **CGNAT** podeu trucar i demanar una **IP pública única** per al vostre rúter. Normalment aquesta IP no serà fixa i canviarà cada setmana. Si la companyia no us pot treure del **CGNAT**, insisteix i informa't sobre la situació del **CGNAT** al teu país. Una altra opció és trucar a altres companyies per preguntar si ofereixen serveis sense **CGNAT** i, en cas afirmatiu, considerar canviar de proveïdor.

Alternatives al CGNAT- els Serveis Ceba: una altra opció que ens han plantejat algunes companyes és configurar un *Onion Service* a la xarxa Tor. Això us permet evitar el **CGNAT**, però només les persones connectades a la xarxa Tor podran veure la vostra web. A més, l'**URL** de la web serà similar a això: **ww6ybal4bd7szmgncyruucpgfkqahzdd.onion**. Aquest mètode també elimina la necessitat d'un domini i augmenta la privacitat. Aquí trobareu un tutorial sobre com configurar un **Servei Ceba:** https://community.torproject.org/es/onion-services/setup/.

Com us hem dit abans, aquesta guia està pensada per muntar una servidora web amb una connexió casolana. Pot ser la de la casa d'alguna de vosaltres, pero tingueu en compte que això comporta riscos. Amb l'adreça IP pública es pot identificar el vostre proveïdor d'Internet (**ISP**), que alhora disposa de dades personals com el nom, l'adreça postal, el número de telèfon, etc., de la persona titular del contracte. La protecció d'aquestes dades depèn de la companyia que hàgiu contractat i de la legislació del vostre país.

10.2 Assignar una adreça IP local fixa a la servidora

Abans d'acabar la configuració del router, és important assignar a la nostra servidora web una **adreça IP local fixa**. Això ens permetrà "localitzar-la" sempre amb la mateixa IP, ja que aquesta sovint canvia cada cop que connectem el dispositiu a la xarxa local.

Hi ha diverses maneres d'assignar una adreça IP local fixa a un dispositiu. Una de senzilla és accedir a l'apartat corresponent de la configuració del rúter i fixar una adreça IP en funció de l'adreça **MAC** del dispositiu. **Compte!** No li assigneu una adreça IP que ja estigui sent utilitzada per un altre dispositiu, ja que això generaria un conflicte.

Alguns rúters casolans permeten triar directament el nom del dispositiu i assignar-li una adreça IP. En altres casos, haureu de trobar l'adreça **MAC** de la targeta de xarxa de la Raspberry Pi. Per fer-ho, podeu executar el següent comandament:

```
ip a
```

I localitzar un conjunt de 6 valors en hexadecimal, com ara: **b8:27:eb:20:a0:9f** (aquest és només un exemple; cada adreça **MAC** és única per a cada targeta de xarxa).

Assegureu-vos que esteu mirant la secció de la interfície cablejada (normalment anomenada **eth0**). Allà també podreu veure l'adreça IP. No la confongueu amb l'adreça **MAC** de la interfície sense fils, que sol anomenar-se **wlan0.**

10.3 Configurar una xarxa DMZ

Ara el nostre objectiu és configurar el rúter perquè la nostra servidora sigui accessible des d'Internet i no només des del salo de casa o de la comunitat. Per fer això, cal obrir alguns ports. **Però, com? És segur per a la resta de dispositius connectats a la mateixa xarxa?** Vegem com fer-ho sense posar-nos en risc.

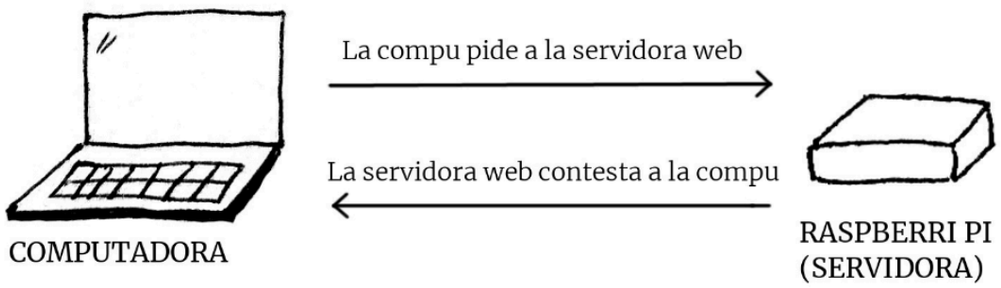
Per "obrir aquestes portes" del rúter una opció recomanada és configurar una **xarxa DMZ** (les sigles signifiquen "zona desmilitaritzada" en anglès) al router. Això permet que qualsevol petició externa que rebi el router i consideri "desconeguda" sigui dirigida a l'adreça IP local que assignem (en aquest cas, la de la nostra servidora amb IP local fixa).

Amb "Petició externa desconeguda" ens referim a una connexió que arriba d'Internet i que cap dispositiu de la xarxa local ha sol·licitat. Aquesta situació és oposada a la de navegar per un lloc web des d'un ordinador: en aquest cas, les respostes a les peticions

són considerades "conegudes" pel rúter.

Una servidora web necessita poder acceptar **peticions desconegudes** que vinguin des de fora per poder oferir els continguts web allotjats en ella. Això és el que s'entén per "servir" una web: la servidora està esperant aquestes peticions desconegudes per respondre-les.

Una xarxa **DMZ** aïlla els altres dispositius de la xarxa interna d'aquestes connexions desconegudes i redirigeix el trànsit cap a la servidora. Potser us ajuda aquest diagrama sobre peticions web:



En la **primera connexió**, la servidora rep una connexió "**desconeguda**", una petició que ella no ha sol·licitat. En la **segona connexió**, l'ordinador rep una connexió **coneguda**, que és la resposta a una petició que ella mateixa ha fet.

Al nostre rúter, la configuració de la xarxa **DMZ** es troba a l'apartat "**S**eguretat > **S**ecurity" o a "**C**onfiguració avançada". Aquí, s'activa la xarxa **DMZ** i s'introdueix el "**LAN** host" (dispositiu de la xarxa local) al qual es redirigiran les connexions externes desconegudes.

Per fer-ho, hem d'introduir l'adreça **IP** local de la servidora. **S**i no la recordeu, podeu obtenir-la amb aquest comandament:

hostname -I

Si no voleu configurar una xarxa **DMZ**, podeu utilitzar *port forwarding* per fer que la servidora sigui accessible des d'Internet. Aquesta opció també dirigeix les connexions no sol·licitades cap a una adreça **IP** específica. **C**aldrà configurar-lo per als ports 80, 443 **I** per al que hàgiu triat per a connexions **SSH** (en el nostre cas, el 2251, per exemple). Això ens ha donat bastants mals de cap així que et recomanem la xarxa **DMZ**

Per comprovar que la configuració ha funcionat, obriu el navegador i escriviu l'adreça **IP pública** assignada al rúter. **S**i tot està configurat correctament, hauríeu de veure la vostra pàgina web. Espectacular!

10.4 Gestionar DNS dinàmics

Com que la nostra xarxa és casolana, el nostre proveïdor de serveis d'Internet no ens assigna una adreça IP pública fixa per al **rúter**. És a dir, un dia ens connectem amb una IP, la setmana següent amb una altra, etc. Això passa perquè no hi ha prou adreces IP públiques per a tots els **rúters** que hi ha al món. Si voleu una IP pública fixa, haureu de pagar un extra al proveïdor de serveis d'Internet. Els **servidors** grans, com els que pertanyen a institucions o empreses, solen pagar per aquest servei.

En aquest cas, aquelles persones que volguessin visitar la nostra web haurien de conèixer la nostra adreça IP de la setmana. Per solucionar-ho, necessitem un servei de **DNS dinàmic** que ens permeti vincular un domini amb una adreça IP pública dinàmica, com la que tenim amb una connexió casolana. Aquest servei fa un seguiment dels canvis de l'adreça IP pública del **rúter** i actualitza la vinculació amb el nostre domini.

El servei que vincula adreces IP públiques amb noms de domini es diu **DNS** (**D**omain **N**ame **S**ystem). Hi ha servidors **DNS** que proveeixen aquest servei. Però primer necessitem un domini.

Conseguir un domini:

Per procedir a vincular un domini amb l'adreça IP dinàmica, haurem de comprar un domini. Si no teniu diners per comprar-ne un de propi, podeu optar per serveis gratuïts com https://www.noip.com, que ofereixen **DNS** dinàmic i proporcionen subdominis com per exemple http://hola-amiga.noip.com o http://hola.ddns.net. A la pròpia pàgina de NoIP hi ha molta documentació sobre com utilitzar el servei. La única desavantatge de la versió gratuïta és que cal renovar-la cada mes (però ens avisen per evitar que se'ns oblidi). Tot i això, pensem que pot generar problemes quan intentem configurar els certificats **SSL/TLS** amb **Let's Encrypt**.

Nosaltres vam comprar un domini de nivell superior **.red**. Abans hi havia poques extensions de domini, com .com, .edu, .gov, .org, .net, etc., i ara n'hi ha moltes més. Podeu veure una llista completa a la pàgina de Wikipedia List of Internet top-level domains. Existen molts llocs per comprar dominis: varien en preu, condicions, prestacions, etc. Si volem comprar un domini de país (com .gt, .es, .py, etc.), hem d'acudir a les agències nacionals. Per a dominis genèrics, podem buscar qualsevol altre servei. Alguns projectes activistes/ètics que ofereixen dominis són: 1984hosting.com, greenhost.net o gandi.net.

Hem de tenir en compte que quan comprem qualsevol domini ens demanaran dades personals de contacte: nom, adreça postal, correu electrònic, etc. Aquests dades (conegudes informalment com **'WHOIS'**) seran públiques, a no ser que sol·licitem que no es mostrin. Algunes companyies cobren per aquest servei i altres no. És important fer una **avaluació de riscos** per saber quines dades proporcionar i com protegir-les.

També serà necessari pensar com fer el pagament. Alguns projectes com 1984hosting.com o njal.la permeten fer la compra amb criptomonedes en lloc d'utilitzar una targeta de crèdit. Fins i tot qwass.com accepta Faircoins, una criptomoneda ètica. Això ens pot ajudar a protegir la nostra privadesa.

Per comprar el domini, haurem de crear un perfil per accedir al **panell de control**, on podrem gestionar una sèrie de paràmetres que veurem més endavant. **Utilitzeu una bona contrasenya i guardeu-la al vostre gestor de contrasenyes!**

Recomanem l'ús de gestors de contrasenyes com **KeePassXC**: us ajudaran a generar contrasenyes fortes, tenir-les ordenades i emmagatzemar-les de manera xifrada.

DNS dinàmics

Després de buscar serveis de **DNS** dinàmics, ens van recomanar el servei de **Hurricane Electric** (https://he.net). Per poder utilitzar els seus serveis de **DNS**, cal crear-se un compte a https://dns.he.net on podreu configurar els registres **DNS**. Un cop us hagueu donat d'alta, aneu al panell d'administració del vostre domini a la web on l'heu comprat i busqueu l'opció per configurar els servidors **DNS**. Aquí, haureu d'introduir els **DNS** de **he.net:**

- **ns1.he.net**
- **ns2.he.net**
- **ns3.he.net**
- **ns4.he.net**
- **ns5.he.net**

Potser caldrà esperar uns minuts fins que la configuració es sincronitzi. Després, torneu al panell de control de **dns.he.net** i afegiu un nou registre de tipus **A** amb els següents valors:

- **name: nuestrodominio.net**
- **IPv4 Address: 1.1.1.1**
- **TTL: actualització del registre en segons**

(podeu posar 300 - 5 minuts)

Marqueu la casella "Enable entry for dynamic DNS"

Un cop afegit el registre, feu clic sobre el símbol de les fletxes en cercle (com d'actualitzar). S'obrirà una finestra de diàleg demanant una contrasenya per al registre. Poseu-la, confirmeu-la i accepteu. **Aquesta contrasenya cal guardar-la al vostre gestor de contrasenyes!**

Atenció! Posem una adreça IP de prova (1.1.1.1) per poder comprovar més fàcilment que es canvia a la correcta. Seguiu llegint!

Configurar ddclient

Ara tornem a la Raspberry Pi per configurar el **DNS** dinàmic. Ho farem amb un programari anomenat **ddclient**, que es comunicarà amb el servei **DNS** dinàmic de he.net i li enviarà la nostra nova adreça IP cada vegada que canviï. Per instal·lar ddclient, executeu el següent comandament:

```
sudo apt install ddclient
```

Per configurar-lo, obriu el fitxer de configuració amb:
sudo nano /etc/ddclient.conf

I escriviu la configuració estàndard següent:

```
pid=/var/run/ddclient.pid
protocol=dyndns2
ssl=yes
use=web
server=dyn.dns.he.net
login=nuestrodominio.net
password='contraseña'
nuestrodominio.net
```

En el camp *login* heu de posar el domini que heu comprat, i en el camp *password* la contrasenya assignada al registre A. És important posar-la entre cometes simples. A la darrera línia, torneu a posar el domini.

A continuació, configureu ddclient perquè funcioni com un **daemon** (és a dir, com un servei que corre contínuament). Per fer-ho, editeu el fitxer:

```
sudo nano /etc/default/ddclient
```

Canvieu les opcions perquè quedi així:

```
run_dhclient="false"
run_ipup="false"
run_daemon="true"
daemon_interval="300"
```

Finalment, per aplicar els canvis, reinicieu el servei amb: **sudo service ddclient restart**

Per veure si la direcció IP s'ha actualitzat correctament o si hi ha errors, executeu:

sudo ddclient -daemon=0 -debug -verbose -noquiet

També podeu revisar els logs amb:

cat /var/log/syslog* | grep ddclient

Si tot està configurat correctament, haureu de veure els canvis a la pàgina de **dns.he.net**. Si actualitzeu la pàgina, al registre A (A Record), on abans deia **1.1.1.1**, ara hauríeu de veure la **IP pública** de la vostra connexió casolana. Per comprovar que la vostra IP està associada al vostre domini, podeu fer un **ping** des de qualsevol terminal:

ping nuestrodominio.net

Tingueu en compte que pot trigar uns minuts perquè es faci efectiva l'actualització, ja que té un **TTL** de **300 segons** (5 minuts). Així que espereu aquest temps abans de preocupar-vos.

Si després d'un temps creieu que no s'està actualitzant correctament la IP, podeu esborrar la caché del servei:

sudo mv /var/cache/ddclient/ddclient.cache /var/cache/ddclient/ddclient.cache.old

Després, torneu a reiniciar el servei i mireu els **logs. Atenció!** No us poseu a reiniciar mil vegades, perquè el servei de **DNS** us pot "banejar" per enviar massa informació ;).

Si voleu afegir diversos dominis (per exemple, si teniu més d'una web), haureu d'afegir els paràmetres per al **segon domini** a la configuració de **ddclient**. Així quedaria el fitxer de configuració **/etc/ddclient.conf:**

```
pid=/var/run/ddclient.pid
protocol=dyndns2
ssl=yes
use=web
server=dyn.dns.he.net

#Primer domini
login=labekka.red
password=xxxxxxxxxxx
labekka.red

#Segon domini
login=midominio.red
password=xxxxxxxxxxx
midominio.red
```

A nosaltres ens va ajudar molt consultar aquesta informació:

- https://www.bidon.ca/en/random/2011-06-16-using-dynamic-dns-feature-dnshenet
- https://joatbloginterim.wordpress.com/2013/06/18/setting-up-ddclient-on-the-raspberry-pi/

Recordeu que partir d'ara, per connectar-vos a la servidora per **SSH**, podeu executar el següent comandament des de qualsevol ordinador connectat a Internet:

sudo ssh -p 2251 pi@nuestrodominio.net

11. Configurem Let's Encrypt

Perquè el trànsit de la nostra web vagi xifrat a través del protocol **HTTPS,** necessitem configurar un **certificat SSL/TLS**. Aquests certificats poden ser de pagament i, de vegades, complicats de configurar a Apache. No obstant això, amb l'eina **Certbot** (https://certbot.e.org/), desenvolupada per l'organització **EFF,** podem configurar un certificat de **Let's Encrypt** (https://letsencrypt.org/) de manera senzilla i gratuïta.

Abans de procedir, cal que hagueu completat l'apartat anterior: aconseguir un **domini** i tenir els **DNS** dinàmics ben configurats. A la pàgina de https://certbot.eff.org/ trobareu instruccions per configurar-ho (tot i que està una mica desactualitzada, la veritat). De fet, diu que cal configurar els backports per poder descarregar **Certbot,** però hem comprovat que no és necessari, així que no cal fer-ho.

Per instal·lar Certbot, executeu aquest comandament:

sudo apt install certbot python-certbot-apache

Un cop instal·lat Certbot, només haureu de seguir els passos de la configuració que us sol·licita. Us demanarà que introduïu el domini de la vostra servidora i un correu electrònic. Comencem la instal·lació executant l'assistent:

sudo certbot --apache

L'assistent de configuració també ens preguntarà si volem redirigir les peticions HTTP a HTTPS. En el nostre cas, direm que sí, seleccionant el número 2 i prement Enter. Si el domini no està correctament associat a una adreça IP, l'assistent no completarà la generació dels certificats.

Els certificats generats es guarden a **/etc/letsencrypt/live/domini/.** Només duren 3 mesos, però es renoven automàticament. Si voleu comprovar que les renovacions funcionen, podeu executar:

sudo certbot renew --dry-run

--dry-run és un modificador molt útil per provar un comandament: l'executa sense fer-ho realment i ens retorna el resultat de què passaria si el féssim. Mola eh? Ojalà existís alguna cosa així per a la vida humana.

Ara podeu obrir el navegador, escriure el vostre domini i comprovar que el certificat està configurat. Veureu un candadet verd a la barra de navegació i que la connexió es fa a través del protocol **HTTPS**. També podeu comprovar que la redirecció de **HTTP** a **HTTPS** funciona, simplement eliminant la **s** de l'**URL** i veient com es redirigeix automàticament.

Si teniu una Raspberry Pi vella, potser trobareu alguns problemes. A nosaltres ens va passar amb una Raspberry Pi 1 Model B+ perquè el Certbot que es troba als repositoris de Debian Stretch (la versió de Debian 9 sobre la qual està basat Raspbian Stretch) no és compatible amb la seva arquitectura. Aquí trobareu algunes orientacions per a aquests casos: https://community.letsencrypt.org/t/certbot-on-rasp-bian-illegal-instruction/15813. Però nosaltres no vam aconseguir configurar-ho i vam haver de fer-ho amb una Raspberry Pi més nova.

Una altra opció és utilitzar **certbot-auto**, que es proposa per a qualsevol sistema operatiu quan no hi ha una versió específica desenvolupada per aquest sistema: https://certbot.eff.org/lets-encrypt/pip-apacheAmb aquest últim pas, ja tenim configurats els **certificats SSL/TLS**.

Felicitats! Aquesta part de la guia ha tingut molts passos, però són **extremadament rellevants** per poder habitar Internet sense exposar-nos. I encara que ara estem més protegides, no hem de confiar-nos. Aquesta és una tasca **diària** de revisió, actualització i proves permanents. Al cap i a la fi, encara que sigui una màquina, la nostra **servidora** també cal cuidar-la.

Us recordem la importància de documentar el vostre procés: anoteu què heu fet, com ho heu fet, amb quines dificultats us heu trobat i com les heu resolt. Una espècie de **diari** que registri totes les decisions que hem pres al llarg del procés. Així podrem reconstruir-lo, multiplicar-lo i tornar a ell cada cop que sigui necessari.

Us sembla si ara desenvolupem la nostra pàgina web?

Som-hi <3!