

Etherpad editor texto colaborativo en línea

Comenzamos el 2017 con un breve tutorial sobre **Etherpad**, específicamente la versión ligera o **Etherpad Lite**, la cual es un editor de texto colaborativo en línea sin mayores pretensiones de seguridad y/o registro pero sencilla y poderosa en concepto. Veremos como instalarla, utilizarla y hasta revisaremos de forma sencilla cómo funciona, acompañenos por favor 8-).

Introducción.

Mucho se habla de las "nubes" en forma metafórica y siempre hemos sido un tanto excépticos al uso -y abuso- de dicho término. Somos sobre todo pragmáticos pero también cautelosos en cuanto a la tecnología que almacena nuestros datos en internet y "prometen el oro y el moro" a diestra y siniestra pero he aquí que nos llamó poderosamente la atención la herramienta Etherpad por su aparente sencillez -luego veremos muy bien los detalles "de debajo del [capó](#)".

En los últimos tiempos hemos visto el surgimiento de aplicaciones en línea (documentos de texto, hojas de cálculo y hasta edición de imágenes) que por una tarifa mensual alquilamos el almacenamiento de los ficheros -[de algo tenemos que vivir los programadores](#), comemos y tenemos familias que mantener- pero tal vez no necesitemos de una herramienta tan sofisticada para nuestros propósitos. He aquí que interviene Etherpad: podremos tener "listas por hacer", documentos, declaraciones, ENTREVISTAS en línea con resaltado sobre lo que escribe cada participante identificado con diferentes colores y podremos embutirlo en cualquiera de nuestras aplicaciones con tan solo incluir una ventana al navegador web que gustéis (lo probamos con Mozilla Firefox y Chromium). Esencialmente eso es lo que hace Etherpad, después que lo instalemos ampliaremos en mayores detalles, no muchos debido a la aparente simplicidad de la aplicación.

La historia de Etherpad está bien resumida en el siguiente párrafo: la aplicación fue lanzada el 19 de noviembre de 2008 por David Greenspan, Aaron IBA y J. D. Zamfirescu (los dos últimos solía trabajar para Google). Más tarde se les unieron -los también empleados por Google- Daniel Clemens y el diseñador David Cole. El sitio original fue llamado [etherpad.com](#) (hoy en día el enlace redirige hacia etherpad.org). El 4 de diciembre de 2009 Google compró AppJet, la compañía que posee el software y se anuncia que esta solicitud de colaboración estará disponible bajo licencia Open Source. El 17 de diciembre de 2009, Google libera los códigos fuentes de Etherpad bajo licencia Apache v2.

<https://twitter.com/EtherpadOrg/status/706409752109174784>

En esta entrada evaluaremos la [versión Etherpad 1.6.1 disponible tanto para GNU/Linux](#) como para [sistemas operativos privativos](#). [Para compartir vía Torrent la versión disponible es la 1.4.1.](#)

Etherpad Lite fue reescrita completamente en un servidor JavaScript utilizando Node.js y del lado

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

del cliente también utiliza javaSript en un 99%.

Los mismos programadores, que antes trabajaron para Google, destacan por Twitter la versatilidad de Etherpad y se erigen como competencia directa al servicio "Google Docs":

<https://twitter.com/EtherpadOrg/status/703905051497140224>

La última versión también está alojada en GitHub.com y está amparada por la licencia [Apache 2](#).

Ambiente de trabajo.

Nosotros utilizamos un ordenador con Ubuntu 16.04 en una red de área local que configuramos por DHCP que nos otorga la siguiente dirección IPv4 fija: 192.168.1.47, aparte de tener derechos de administrador "root" eso es todo lo que necesitamos, ¿sencillo, cierto?

Proceso de instalación.

Abrimos [una ventana terminal](#) e introducimos el siguiente comando:

```
sudo apt install git curl python libssl-dev pkg-config build-essential
```

Cada una de las palabras a la derecha de "apt install" respresenta un sfotware en particular que no describiremos detalladamente pero por acá hemos escrito sobre [Git](#) y [Python](#), por si acaso queréis ampliar vuestros conocimientos pero para el caso que nos ocupa no es necesario saber usarlos -pero al menos tener noción de que existen-.

Como véis nosotros ya teníamos instalado lo necesario debido a nuestro trabajo de programación en otros asuntos (esto es lo que denominamos entorno de trabajo, *se comparten muchos elementos, cuidado de poner muchos proyectos en una sola computadora por razones de estabilidad y uso de versiones compartidas*).

<https://twitter.com/ks7000/status/817731452850814981>

Ahora procederemos a instalar [Node.js 6.9.2](#) con la siguiente orden:

```
wget https://nodejs.org/dist/v6.9.2/node-v6.9.2-linux-x64.tar.xz
```

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

Una vez descargado con **wget** los archivos comprimidos procedemos a extraerlos con el comando **tar** en la carpeta donde lo descargamos:

```
tar xJf node-v6.9.2-linux-x64.tar.xz
```

Luego creamos un directorio y movemos los archivos pero con derechos de administrador "root":

```
sudo mkdir /opt/nodejs/ && sudo mv node-v6.9.2-linux-x64/* /opt/nodejs
```

En la imagen anterior, por error, no usamos la orden "**sudo mv**" y eso ocasionó los mensajes de "no se pudo mover" es por ello que casi no usamos el concatenador "&&", generalmente ejecutamos paso por paso, *pero bueno, errar es de humanos y reconocemos nuestro fallo*, a la final completamos el proceso en una segunda línea.

Tras la pequeña falla subsanada, agregamos los enlaces simbólicos hacia nuestro perfil de usuario en el ordenador:

```
echo "PATH=$PATH:/opt/nodejs/bin" >> ~/.profile
```

Si todo sale bien pues **felicitaciones**, tenemos instalado Node.js en nuestra máquina de pruebas.

Instalando el código fuente de Etherpad.

En idioma inglés a esto lo llaman "instalar los (archivos) binarios", nosotros simplistas siempre pues lo llamamos código fuente o más simple aún: fuente de Etherpad. Para ello haremos lo siguiente:

1. Crearemos un directorio llamado **etherpad** en la carpeta **opt**.
2. Cambiaremos el propietario de la carpeta con [chown](#) hacia nuestro nombre de usuario [whoami](#) aprovechando las cualidades del bash y el uso de variables de paso entre comandos.
3. Cambiamos al directorio creado ya sin credenciales de administrador "root" porque la

carpeta nos pertenece ahora.

4. utilizamos el comando **git** para descargar el código fuente de Etherpad.

```
sudo mkdir /opt/etherpad sudo chown -R $(whoami).$(whoami) /opt/etherpad
cd /opt/etherpad git clone git://github.com/ether/etherpad-lite.git
```

Ejecutando por primera vez Etherpad.

Para comenzar a usar Etherpad ("levantar el servidor") pues ejecutamos un *archivo de procesos por lote o bash* llamado "[run.js](#)" desde donde se verifican las variables de entorno (esencialmente nos prohíbe ejecutarlo con derechos de administrador "root" **recordad que es un servicio web expuesto a la internet y a todo el mundo**) y allana el camino para ejecutar a su vez a otro bash, [installDeps.sh](#), que hace las siguientes comprobaciones:

1. Se asegura que NO se haya llamado desde el directorio **bin (en nuestro ordenador de ejemplo nos quedó instalado en la siguiente vía: "/opt/etherpad/etherpad-lite")**.
2. Comenzamos a usar el comando [hash](#) que devuelve cero si encuentra la vía donde están instaladas las respectivas aplicaciones (¿recuerdan lo que comentamos sobre agregar enlaces simbólico a la vía **PATH** al fichero **/profile**)
3. En el caso que estemos ejecutando sobre SunOS necesitaremos instalar el comando **ggrep** -no se aplica a nuestro ordenador con Ubuntu-.
4. Verifica que tengamos instalado [curl](#) el cual usaremos para descargar lo que necesitamos de la internet.
5. Verifica si tenemos **node.js** instalado.
6. Verifica si tenemos [npm](#) instalado -y su versión-. Npm nos permite manejar cientos de miles de aplicaciones escritas para JavaScript.
7. Verifica la versión de node.js instalada (0.10.x o superior).
8. Confirma el nombre del archivo "**settings.json**" (pronto lo abriremos y editaremos).
9. De no conseguir "**settings.json**" pues copia una plantilla por defecto para poder comenzar a trabajar.
10. De nuevo tomamos a **npm** y le indicamos que actualice o instale toda nuestra "paquetería" (**este comando puede tomar algo de tiempo dependiendo de vuestra máquina y vuestra velocidad de conexión al internet**).
11. Verificamos que la versión **jQuery** que tengamos sea mínimo 1.9.1 *de no ser así utilizamos [curl](#) para descargarla*.
12. Importante: en la carpeta **"/var"** donde tengamos instalado Etherpad (para nuestro ejemplo en **"/opt/etherpad/etherpad-lite/var"**) reposan los archivos que comienzan con la palabra **"minified"** que esencialmente contienen los guiones JavaScript que corren del lado del cliente. SUPONEMOS que son borrados y creados de nuevo cada vez que se ejecuta

Etherpad porque se trata de reunir todas las piezas y entregarlas al cliente para que se ejecuten allá. Cada archivo contiene una licencia de uso Apache 2 e imploran ayudar a comentar -y ayudar- a mantener el código. **Esta parte garantiza que cada que vez que "reiniciemos" a Etherpad sean instaladas y "compiladas" las intrucciones al cleinte con las nuevas versiones.** También en esta carpeta reposa la base de datos **"dirty.db" -manejada con SQLite-** la cual es para propósitos de desarrollo y pruebas (lo que enseñamos nosotros en esta vuestra página web) pero que para entornos "de producción" recomiendan sustituir con **mySQL**.

13. Se asegura que tenga los guiones **.js** y **.css** "limpiecitos" antes de comenzar a ejecutar.

Nos tomamos el "atrevimiento" de copiar el archivo **installDeps.sh** tal cual se muestra al día sábado 07 de enero de 2017 y le colocamos, como valor agregado, instrucciones al usuario en idioma castellano. **Si vosotros queréis contribuir al código fuente central cread una cuenta en GitHub y haced una divergencia de código "fork" en vuestra cuenta, programáis y modificáis y hasta quien sabe, a futuro podéis proponer cambios a los programadores originales para que lo incorporen a la rama oficial.** Pero por ahora nosotros lo que queremos es enseñaros unos pocos archivos y no vale la pena copiar todo el código, por eso GitHub tiene una sección llamada **Gist** (tiza) para estos casos menores y rápidos, especial para ejemplos pequeños, mirad como queda nuestro trabajo:

Ahora si que se ejecuta el guión **server.js** el cual es el que consideramos "compila" y "levanta" el servidor pero mejor os colocamos la cabecera del archivo para que tengaís una noción al respecto:

```
#!/usr/bin/env node /** * This module is started with bin/run.sh. It sets up a Express HTTP and a Socket.IO Server. * Static file Requests are answered directly from this module, Socket.IO messages are passed * to MessageHandler and minified requests are passed to minified. */ /* * 2011 Peter 'Pita' Martischka (Primary Technology Ltd) * * Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS-IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License. */
```

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

Como vemos, debe ser llamado desde **run.sh** quien a su vez -ya explicamos- actualiza y borra datos anteriores con **installDeps.sh** y ejecuta el servicio en sí. Ahora tomad un cafecito y volvéis para mostraros cómo funciona del lado del cliente, con lo que nos toparemos todos los días.

Ejecutando del lado del cliente.

Nuestro servidor de pruebas está ubicado en la dirección IPv4 fija en nuestra red de área local con los números 192.168.1.47 y nos conectaremos con el puerto 9001 el cual es por defecto (de hecho ya tenemos un servidor web Apache2 escuchando en los puertos 80 y 443). Veremos lo siguiente:

Por otra parte nos fijamos en la ventana terminal desde donde lanzamos Etherpad y observamos cierta actividad.

En esta última imagen apreciamos la versión que tenemos instalada y *la advertencia que no tenemos configurados al usuario "admin" y su contraseña (más adelante lo modificaremos)* y también vemos que se ha creado un archivo llamado "prueba" con un nombre aleatorio y la dirección IP del navegador que tenemos conectado.

Esto es lo que veremos, un mensaje de bienvenida, en inglés (ya vamos a cambiar la configuración) y podemos desde ya el comenzar a escribir lo que necesitemos o queramos.

Desde otra computadora ubicada en la misma red de área local nos conectamos y de una vez en el registro o bitácora nos mostrará a cual archivo se conectó y su dirección IP así como la dirección IP.

En ambas computadoras procedemos a identificarnos para que podamos visualizar en diferentes colores lo escrito por cada uno de los participantes.

Abajo a la derecha (esquina inferior derecha) hallaremos el ícono para iniciar una conversación aparte del documento que estemos editando. No consideramos explicar mayor cosa porque sabemos que los usuarios son expertos en esto de los "chats" ;-)

Si hacemos click en la esquina superior derecha, en el ícono de una rueda dentada, abriremos las opciones de configuración para que nos sintamos a gusto en nuestro entorno:

- Mostrar siempre la conversación en pantalla.
 - Mostrar conversación y usuarios.
- Ver lo que escribe cada usuario con su respectivo color.
- Mostrar los números de línea (útil si un documento es extenso y queremos mostrar específicamente algo a otros usuarios).
- Podemos elegir si queremos escribir de derecha a izquierda (idiomas como el árabe, ver siguiente punto).
- El menú desplegable para cambiar el idioma, hay muchos, incluido el "español" -castellano en realidad-.

Configurando Etherpad para su uso del lado del cliente.

Imaginemos ahora que ya tenemos "montado" nuestro servidor en producción y queremos compartir el enlace al documento que estamos editando, pues hacemos click en el ícono "" y copiamos el enlace para enviarlo por correo electrónico, Twitter, etc.

Visualizando los cambios realizados.

Muy útil poder visualizar los cambios realizados, Etherpad conserva un historial extremadamente detallado, lo notamos mucho cuando se importan documentos al trabajo que estamos realizando (ver el próximo punto). Para ello hacemos click en el ícono de reloj de manecillas y podremos adelantar, atrasar y cada cambio hecho se mostrará en pantalla.

Importando documentos.

Esta opción no es muy lógica que digamos pero bueno, muchas veces podemos ahorrar tiempo "copiando y pegando" o, como en este apartado, importando documentos. Las instrucciones dicen que deben ser en formato .html .txt o formato Etherpad pero si instalamos AbiWord podremos "soportar" más formatos de archivo. Para este proceso damos click en las flechas encontradas y "subimos" el documento deseado.

Guardando documentos.

Con solo hacer click en la estrella de cinco puntas el documento será guardado en la base de datos "dirty.db" para una próxima sesión (recordad que podemos configurar para que trabaje con MySQL). Si queremos llevar nuestro trabajo a otro servidor o simplemente respaldar pues lo que tenemos que hacer es copiar y/o comprimir (con fecha y hora en curso) dicho fichero. Esto es importante si tenemos la necesidad de apagar o reiniciar nuestro servidor (cosa extraña en

ambiente GNU/Linux pero sucede). *Ah, por cierto, en el ambiente Etherpad cada documento es denominado "pad".*

Bitácora de registros de eventos en el lado del servidor Etherpad.

Cada usuario conectado o desconectado, nuevo o ya registrado, lo veremos en la ventana terminal de donde lanzamos nuestro servidor Etherpad. Simplemente mirad y admirad la obra de los programadores de software libre.

Configuraciones avanzadas en Etherpad.

Pues algo que nos sorprendió desde un principio fue que no pedía contraseña para abrir o crear un documento o "pad" como se le denomina por acá. Resulta ser que las configuraciones avanzadas las podemos acceder en la carpeta "/admin". Volvemos a nuestro ejemplo en red de área local:

`http://192.168.1.47:9001/admin`

Una vez allí nos pedirá un usuario y contraseña. Pues simplemente para cambiar las credenciales le damos cancelar y finalizamos el proceso que tenemos corriendo en la ventana terminal donde iniciamos el servidor Etherpad. Luego debemos editar el archivo "**settings.json**". Acá os mostramos cómo "tumbamos" el servidor y comenzamos a editar el fichero.

Al abrir veremos cantidad de configuraciones y si nos ponemos a detallar vemos como importantes las opciones para poder trabajar con protocolo SSL (servidores web seguros https) y para conectar con una base de datos MySQL (aunque también pueden ser redis, mongodb, cassandra, MariaDB, etc) **pero cambiemos las credenciales del administrador y del usuario que vienen por defecto**. Buscamos la siguiente sección y cambiamos los valores.

```
/* Users for basic authentication. is_admin = true gives access to /admin.
   If you do not uncomment this, /admin will not be available! */ /*
  "users": {      "admin": {      "password": "changeme1",      "is_admin": true      },      "user": {      "password": "changeme1",      "is_admin": false      }      }, /*
```

Lo que debemos hacer es descomentar la sección (eliminar "/*" y "*/") y colocamos los valores que

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

queramos, guardamos y "levantamos" de nuevo el servidor con "**run.sh**".

```
"users": {      "admin": {      "password": "etherpad",      "is_admin":  
: true      },      "user": {      "password": "etherpad",      "is  
_admin": false      }      },
```

Arriba vemos los cambios, si queremos agregar otro usuario que tenga derechos de administrador debemos colocar la variable "is_admin" a "true". Una vez hayamos reiniciado el servicio Etherpad navegamos a la página de configuración, introducimos nuestro usuario y contraseña y veremos la siguiente pantalla:

Al hacer click en "settings" podremos volver a modificar el archivo "settings.json" vía web e incluso podemos "reiniciar" en caliente el servidor Etherpad.

Cambiando usuarios y "plugins".

Ya en este punto suponemos que estamos lo suficientemente "empapados" de Etherpad por lo tanto ahora os traemos un vídeo donde rápidamente se realiza lo siguiente:

- Se descarga e instala Etherpad en un sistema operativo privativo.
 - Se cambian las contraseñas de los usuarios.
- Se cambia los "plugins" que queramos (que son bastantes, hay variedad).

¿Qué les pareció la experiencia de aprendizaje?

Fuentes consultadas:

En idioma castellano.

- ["Etherpad, editor de textos web colaborativo en tiempo real para Ubuntu"](#)
by Pablo Aparicio

En idioma francés.

- ["Etherpad"](#) - Wikipédia.
- ["L'application collaborative EtherPad sera proposée en Open Source"](#)

KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

Etherpad-Lite instances francophones:

- <https://pad.lqdn.fr/>
- <https://etherpad.fr/>

En idioma inglés.

- "[Install Etherpad web-based real-time collaborative editor on Ubuntu 16.04 Linux](#)" by Essodjolo Kahanam.
- "[A really-real time collaborative word processor for the web](#)" at Github.
- [Etherpad Foundation's Twitter](#).
- [Node.js 6.x documents](#).
- "[How etherpad-lite, a real time collaborative editor, works?](#)" by Shobhit Garg.