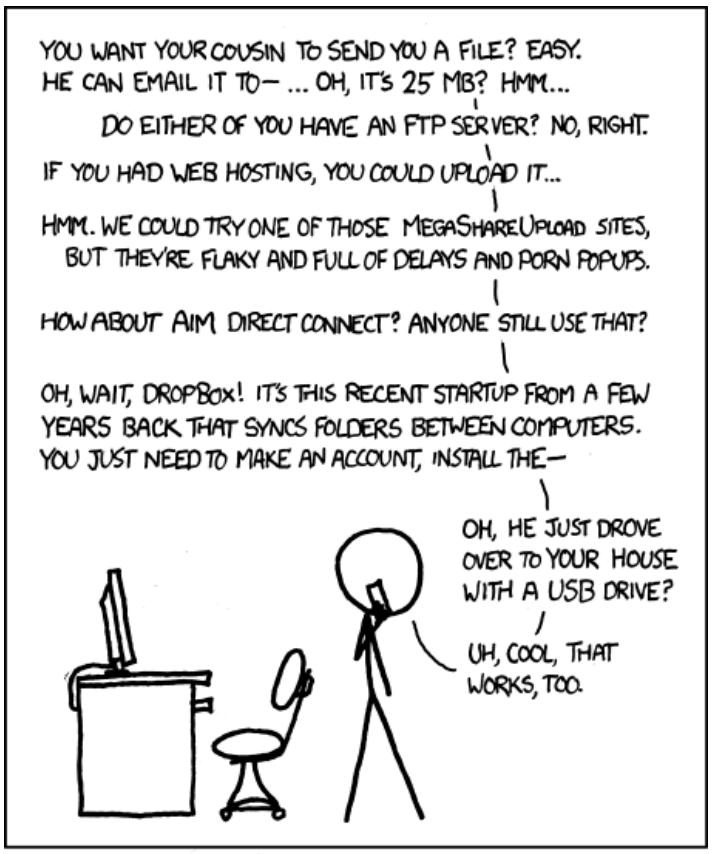
Proyecto 1 - TuMegaBoxDrive.com.ve



I LIKE HOW WE'VE HAD THE INTERNET FOR DECADES, YET "SENDING FILES" IS SOMETHING EARLY ADOPTERS ARE STILL FIGURING OUT HOW TO DO.

Uno de los usos más comunes y convenientes de las redes de computadoras es la transferencia de archivos. Desde los protocolos más simples (como por ejemplo `TFTP`) a los más complejos (por ejemplo `FTP` a secas), la posibilidad de compartir archivos ha sido prevista desde los origenes de la Internet misma. Este caso de uso ha visto su expresión más moderna en la posibilidad de compartir archivos mediante sistemas Web centralizados, con fundamento en el paradigma de Computación en la Nube. Ejemplos de esto son los ubicuos servicios de almacenamiento y compartición de archivos como Mega, Dropbox, Google Drive y ONE Drive, entre muchas otras alternativas.

Sin embargo, una realidad con la que hay que lidiar al usar servicios en la nube es el hecho de que para poder usar el servicio es necesario otorgarle el derecho a examinar nuestros archivos al provedor del servicio. Por diversas razones esto puede ser no deseable para los usuarios, quienes no tendrían más remedio que evitar usar el servcio correspondiente. Podemos hacer algo mejor que esto.

Gracias a las herramientas de seguridad modernas, es perfectamente posible desarrollar un servicio de almacenamiento en la nube capaz de garantizar la privacidad de los usuarios. Combinando técnicas de criptografía simétrica y asimétrica, junto a herramientas de certificación digital, autenticación y firewalls, procederemos entonces a desarrollar nuestro propio sistema seguro de almacenamiento de archivos.

Requerimientos

El sistema a desarrollar debe funcionar como una aplicación Web compuesta por un *backend* y un *frontend*. El *backend* se encargará de las tareas de autenticación de usuarios y almacenamiento de archivos. El *frontend* tendrá como responsabilidad el encriptado/desencriptado de información y la interación con el usuario. Dicho en mayor nivel de detalle, los dos componentes del sistema deben cumplir los siguientes requerimientos:

Backend

- Los archivos y directorios de los usuarios registrados en el sistema deben almacenarse de forma cifrada usando el algoritmo AES-256.
- El *backend* NO debe almacenar ni recibir de ninguna manera la clave AES de los usuarios.
- Las jerarquías de directorios y archivos creadas por los usuarios deben ser ocultas al administrador del *backend*.
- El acceso al servidor para labores de mantenimiento debe realizarse mediante SSH. Este acceso debe ser protegido apropiadamente utilizando las herramientas `iptables` y `fail2ban`.

Frontend

- Para poder acceder a las funcionalidades listadas se debe autenticar al usuario primero. Esto debe realizarse mediante el protocolo OAuth 2.0 utilizando el API de Facebook[1].
- El usuario debe ser tener la capacidad de cargar archivos al sistema y descargarlos posteriormente.
- Se debe presentar una vista que permita al usuario consultar los archivos que tiene almacenados en el sistema, visualizado como una jerarquía de directorios.
- Debe existir la posibilidad de crear, eliminar y mover archivos y directorios mediante la interfaz del sistema.

Todo acceso al servidor debe ser realizado mediante el protocolo HTTPS, utilizando un certificado digital para el servidor correctamente configurado. El certificado digital puede ser auto-firmado u obtenido por medio de Let's Encrypt[2]. No hace falta utilizar certificados de cliente.

Entregables

Para considerar el proyecto completo, debe entregar los siguientes recaudos:

- Una imagen de máquina virtual .vdi (para VirtualBox) con su proyecto instalado y configurado como se indicó anteriormente.
  - El sistema operativo para la instalación es de libre elección.
  - Se recomienda usar OpenBSD 6.5, Debian 9 o CentOS 7.6.
- Un documento llamado README.txt, el cual debe contener su nombre y cédula.
- Un documento llamado INSTALL.txt, el cual debe contener las instrucciones a seguir para poder instalar y configurar un servidor con su proyecto.
- Un reporte en formato PDF de a lo sumo 10 páginas, donde detalle el diseño de su solución, herramientas utilizadas y problemas encontrados durante el desarrollo y como los solucionó.

Los archivos txt requeridos pueden ser escritos con sintaxis Markdown o cualquier otra de su preferencia.

[1] `https://developers.facebook.com/docs/facebook-login`
[2] `https://letsencrypt.org/`

Consideraciones

El desarrollo del proyecto debe hacerse en base a las siguientes consideraciones:

- La fecha de entrega es el día 19 de julio de 2019.
- El proyecto puede realizarse en grupos de a lo sumo dos personas.
- Se permite el uso de cualquier lenguaje de programación, bibliotecas y/o *frameworks*.
- Todos los integrantes de cada grupo deben estar el día de revisión del proyecto. No se calificará a los estudiantes que falten.
- En caso de no poder asistir el día de la revisión, debe notificarlo con antelación y poseer su correspondiente justificativo.
- Las copias serán penalizadas con la nota mínima para todos los involucrados.