

.Debian

銀河系唯一のDebian専門誌

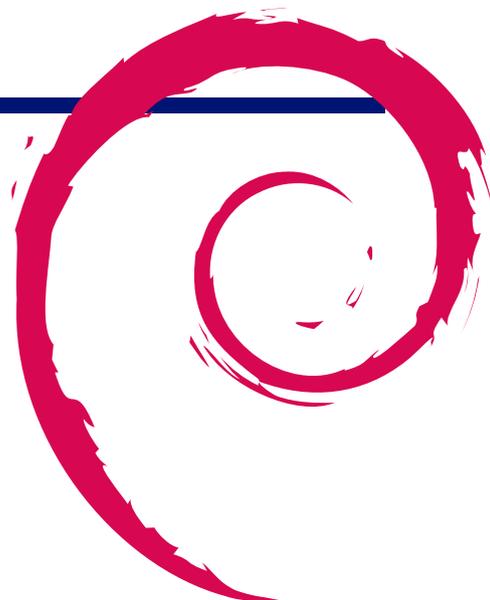
2021年2月20日

Etherpad 特集



会 勉 強 会 の ア ー キ ビ ト

目次	
1	最近の Debian 関連のミーテ ィング報告 2
1.1	2021 年 1 月度 東京エリア・ 関西合同 Debian 勉強会 . . . 2
2	事前課題 3
2.1	dictoss 3
2.2	uwabami 3
2.3	Hiroyuki Yamamoto (yama1066) 3
2.4	NOKUBI Takatsugu (knok) 3
2.5	kenhys 3
2.6	yosuke_san 3
2.7	Kazuhiro NISHIYAMA (znz) 3
2.8	ysaito 3
2.9	yy-y-ja-jp 3
2.10	gdevmjc 3
3	Web ブラウザでメモの同時共 有ができる Etherpad lite の紹介 4
3.1	はじめに 4
3.2	Etherpad と Etherpad lite について 4
3.3	Etherpad lite にたどり着く までの利用変遷 4
3.4	Etherpad lite のインストール 5
3.5	おわりに 8
3.6	参考情報 8
4	メモ 9



1 最近の Debian 関連のミーティング報告

杉本 典充

1.1 2021 年 1 月度 東京エリア・関西合同 Debian 勉強会

2021 年 1 月 16 日 (土) に東京エリア Debian 勉強会と関西 Debian 勉強会の合同でオンラインによる Debian 勉強会を開催しました。参加者は 11 名でした。

セミナーは上川さんによる「Debian での Node.js」および参加者全員参加による BoF「2021 年の活動への抱負」を行いました。

セミナー「Debian での Node.js」では Node.js で使われる javascript の言語仕様の変遷、バージョン番号の見方の説明がありました。その後 Debian で Node.js を使う方法や Debian パッケージで配布する Node.js 関連パッケージのパッケージングルールの説明がありました。ただ、Node.js の開発が活発に行われているがゆえに Debian の安定版で配布する Node.js パッケージは古いバージョンになりがちという問題もあるとのこと。そのため Node.js を Debian で使うには残念ながら Node.js の本家サイトが配布するパッケージをインストールして使うのが無難ということでした。

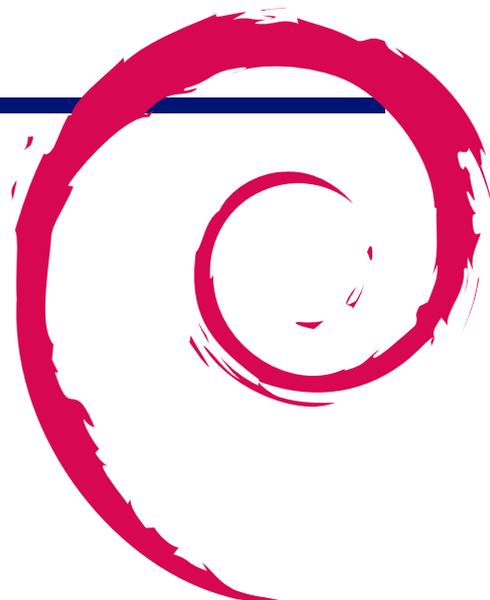
BoF「2021 年の活動への抱負」では 2021 年に何を学習していくとよいかのアイデアを参加者で出し合いました。出てきたアイデアは以下 URL の議事録にまとめました。

https://tokyodebian-team.pages.debian.net/2021-01_tokyodebian_bof.txt

その後、Debian の次期安定版「bullseye」のリリースに向けた進捗の確認、Debian の最近の動向について情報交換を行いました。

2 事前課題

杉本 典充



今回の事前課題は以下です。

1. ModSecurity というソフトウェアを知っていましたか。
2. Etherpad というソフトウェアを知っていましたか。

2.1 dictoss

1. 今回初めて知った
2. 知っており、使ったことがある

2.2 uwabami

1. 知っており、使ったことがある
2. 知っており、使ったことがある

2.3 Hiroyuki Yamamoto (yama1066)

1. 今回初めて知った
2. 今回初めて知った

2.4 NOKUBI Takatsugu (knok)

1. 知っているが、使ったことはない
2. 知っており、使ったことがある

2.5 kenhys

1. 知っており、使ったことがある
2. 知っており、使ったことがある

2.6 yosuke_san

1. 今回初めて知った
2. 知っており、使ったことがある

2.7 Kazuhiro NISHIYAMA (znz)

1. 知っており、使ったことがある
2. 知っているが、使ったことはない

2.8 ysaito

1. 今回初めて知った
2. 今回初めて知った

2.9 yy-y-ja-jp

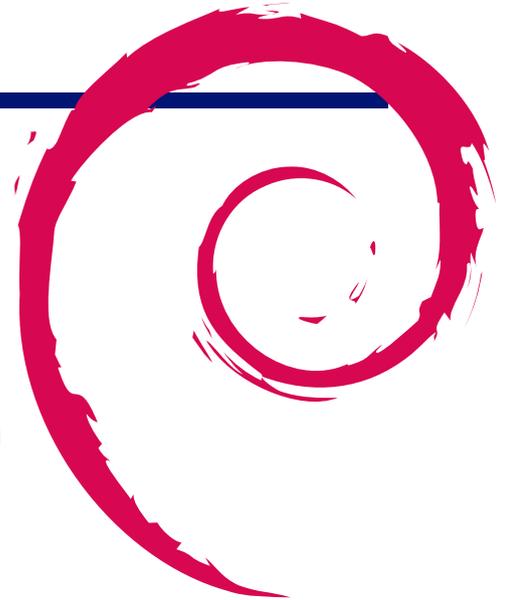
1. 知っているが、使ったことはない
2. 知っており、使ったことがある

2.10 gdevmjc

1. 知っており、使ったことがある
2. 知っているが、使ったことはない

3 Web ブラウザでメモの同時共有ができる Etherpad lite の紹介

杉本典充



3.1 はじめに

Debian 勉強会を進行をするにあたり、参加者が同時にメモの編集ができる環境がほしいと思っていました。Web ブラウザを使って複数人が同時にメモの参照や編集ができるツールである「Etherpad lite」を使ってみました。

3.2 Etherpad と Etherpad lite について

Etherpad (<https://etherpad.org/>) とは、Web ブラウザを使って複数人が同時にリアルタイムでメモの参照と編集ができるオープンソースソフトウェア*1です。

歴史的には(初代の) Etherpad は 2008 年に etherpad.com (現在は etherpad.org へリダイレクトされる) でプロプラエタリなサービスとして登場し、Java と Scala で実装したものでした。その後 2009 年に etherpad.com は Google に買収され、買収の直後に etherpad のソースコードが Apache License の元で公開されてオープンソースとなりました*2*3。

(初代の) Etherpad がオープンソースで公開された後、2011 年 8 月 22 日に JavaScript と Node.js で Etherpad の機能を再実装した「etherpad-lite」の ver 1.0 が登場しました。etherpad の開発を行う The Etherpad Foundation は今後 etherpad-lite の開発に専念することを決めたようで、(初代の) Etherpad は 2013 年に開発を終了しています。なお、etherpad-lite については The Etherpad Foundation の元で現在でも開発が続けられています。

こうした背景があり(初代の)「Etherpad」の開発は既に終了していることから、現在では単に「etherpad」というと「etherpad-lite」を指すようになっていきます*4。

3.3 Etherpad lite にたどり着くまでの利用変遷

だいぶ前となりますが Web ブラウザを使って複数人が同時にメモを編集でき無料で使える Web サービスとして titanpad.com*5 がありました。titanpad.com は便利なサービスのため Debian 勉強会や他の勉強会でも利用されていましたが、2017 年 12 月 31 日をもってサービスを終了しました*6。

*1 <https://github.com/ether/etherpad-lite>

*2 <https://code.google.com/archive/p/etherpad/>

*3 リリース アナウンス <https://web.archive.org/web/20091221023828/http://etherpad.com/ep/blog/posts/etherpad-open-source-release>

*4 <https://etherpad.org/#download> のダウンロード先のリンクは etherpad-lite になっています。

*5 ソースコードのアーカイブはこちら <https://github.com/titanpad/titanpad>

*6 http://blog.titanpad.com/2016/11/shutting-down-titanpad_12.html

その後、Debian 勉強会ではメモの同時編集ツールとして DebConf などでも使われている gobby^{*7} を使っていたが gobby は使いにくいという声（特に debian を使っていない方）もあり、Web ブラウザで利用できるメモの同時編集ツールを再度探していました。

ツールを探していく中で Web サービスではなく自分のサーバにインストールするタイプの「etherpad-lite」があることを知り、最近では Debian 勉強会に導入し BoF で利用しています。

3.4 Etherpad lite のインストール

3.4.1 Node.js のインストール

etherpad-lite のインストール手順は以下 URL に記載されています。

<https://github.com/ether/etherpad-lite#installation>

現在の etherpad-lite-1.8.8 では nodejs-10.17.0 以上の環境が必要であり、以下コマンドで nodejs をインストールします。

```
$ curl -sL https://deb.nodesource.com/setup_14.x | sudo -E bash -
# apt install -y nodejs
```

3.4.2 Etherpad lite のダウンロードと配置

etherpad-lite のプログラムを以下 URL からダウンロードして、/opt に配置します。

```
# wget https://github.com/ether/etherpad-lite/archive/1.8.8.zip
# cp 1.8.8.zip /opt/etherpad-lite-1.8.8.zip
# cd /opt
# unzip etherpad-lite-1.8.8.zip
# ln -s etherpad-lite-1.8.8 etherpad-lite
```

etherpad-lite のプログラムを実行するスクリプト "bin/run.sh" は、実行時にプログラムを置いているディレクトリの配下に "node_modules" ディレクトリを作成して依存ライブラリを自動でダウンロードします。そのため、プログラムを実行するユーザはこのディレクトリに書き込み権限を持つ必要があります。プログラムを実行するユーザとグループとして "etherpad" ユーザ及びグループを作成し、etherpad-lite ディレクトリとファイルのオーナーとグループを変更します。

```
# adduser etherpad
# chown -fR etherpad:etherpad /opt/etherpad-lite-1.8.8
# ls -l /opt/etherpad-lite-1.8.8
合計 180
-rw-r--r-- 1 etherpad etherpad 64 2月 18 22:17 APIKEY.txt
-rw-r--r-- 1 etherpad etherpad 40276 2月 16 02:47 CHANGELOG.md
-rw-r--r-- 1 etherpad etherpad 8217 2月 16 02:47 CONTRIBUTING.md
-rw-r--r-- 1 etherpad etherpad 1681 2月 16 02:47 Dockerfile
-rw-r--r-- 1 etherpad etherpad 11353 2月 16 02:47 LICENSE
-rw-r--r-- 1 etherpad etherpad 849 2月 16 02:47 Makefile
-rw-r--r-- 1 etherpad etherpad 8229 2月 16 02:47 README.md
-rw-r--r-- 1 etherpad etherpad 118 2月 16 02:47 SECURITY.md
-rw-r--r-- 1 etherpad etherpad 64 2月 18 22:16 SESSIONKEY.txt
lrwxrwxrwx 1 etherpad etherpad 7 2月 18 22:10 bin -> src/bin
drwxr-xr-x 6 etherpad etherpad 4096 2月 16 02:47 doc
drwxr-xr-x 2 etherpad etherpad 4096 2月 16 02:47 node_modules
-rw-r--r-- 1 etherpad etherpad 21152 2月 16 02:47 settings.json.docker
-rw-r--r-- 1 etherpad etherpad 19209 2月 16 02:47 settings.json.template
drwxr-xr-x 9 etherpad etherpad 4096 2月 18 22:17 src
-rw-r--r-- 1 etherpad etherpad 51 2月 16 02:47 start.bat
lrwxrwxrwx 1 etherpad etherpad 9 2月 18 22:10 tests -> src/tests
drwxr-xr-x 2 etherpad etherpad 4096 2月 18 22:34 var
```

次に設定ファイル "settings.json" を配置します。設定内容はひとまずデフォルトのコピーとしておきます。

```
# cd /opt/etherpad-lite-1.8.8
# cp -p settings.json.template settings.json
```

etherpad-lite を起動できることを確認してみます。

^{*7} <https://github.com/gobby/gobby/wiki>

```
# cd /opt/etherpad-lite-1.8.8
# sudo -u etherpad bin/run.sh
```

以下の URL のアクセスすると etherpad-lite に接続できます。

<http://localhost:9001/p/debianmeeting20210220>

3.4.3 systemctl で起動できるように service ファイルを配置する

毎回手動で etherpad-lite を起動するのも面倒です。そのため、systemctl から起動できるように設定します。

systemctl で起動できるように以下のような service ファイルを作成します。

```
# vi /etc/systemd/system/etherpad-lite.service

[Unit]
Description=etherpad-lite

[Service]
Type=simple
ExecStart=/opt/etherpad-lite/bin/run.sh
WorkingDirectory=/opt/etherpad-lite
Restart=always
RestartSec=10
User=etherpad
Group=etherpad
Environment=NODE_ENV=production

[Install]
WantedBy=multi-user.target
```

systemctl daemon-reload を実行して追加した service ファイルを systemd に再読み込みさせ、etherpad-lite を起動します。

```
# systemctl daemon-reload
# systemctl start etherpad-lite
# systemctl status etherpad-lite
etherpad-lite.service - etherpad-lite
Loaded: loaded (/etc/systemd/system/etherpad-lite.service; disabled; vendor p
Active: active (running) since Fri 2021-02-19 22:11:05 JST; 4s ago
Main PID: 26451 (run.sh)
Tasks: 13 (limit: 1149)
Memory: 155.0M
CGroup: /system.slice/etherpad-lite.service
        26451 /bin/sh /opt/etherpad-lite/bin/run.sh
        26454 /bin/sh src/bin/installDeps.sh
        26491 npm
```

これで、systemctl で管理しつつ etherpad-lite を起動できるようになりました。

3.4.4 Web サーバへのアクセスを Etherpad lite ヘリバースプロキシする

これまでの手順で起動できるようになった etherpad-lite の状態では、9001 番ポートで待ち受けします。クライアントとなる Web ブラウザから etherpad-lite サーバの 9001 番ポートへ接続する場合に、利用するネットワークやファイアウォール設定などの環境によって接続できない場合があります。

以下 URL を参考に apache2.4 の mod_proxy を使ってリバースプロキシの設定を行い、クライアントからは 80 番ポート (http://) または 443 番ポート (https://) で接続できるようにしてみました。

<https://github.com/ether/etherpad-lite/wiki/How-to-put-Etherpad-Lite-behind-a-reverse-Proxy> まずは apache2.4 をインストールします。

```
# apt update
# apt install apache2
```

apache2.4 の mod_proxy などの利用するモジュールを有効にします。

```
# a2enmod proxy
# a2enmod proxy_http
# a2enmod proxy_balancer
# a2enmod lbmethod_byrequests
# a2enmod rewrite
# a2enmod proxy_wstunnel
```

etherpad-lite の 9001 番ポートへリバースプロキシする設定ファイル ”etherpad-lite.conf” を作成します。

```
# vi /etc/apache2/conf-available/etherpad-lite.conf

ProxyVia On
ProxyRequests Off
ProxyPreserveHost on

<Location /etherpad/>
  ProxyPass http://localhost:9001/ retry=0 timeout=30
  ProxyPassReverse http://localhost:9001/
</Location>
<Location /etherpad/socket.io>
  # This is needed to handle the websocket transport through the proxy, since
  # etherpad does not use a specific sub-folder, such as /ws/ to handle this kind of traffic.
  # Taken from https://github.com/ether/etherpad-lite/issues/2318#issuecomment-63548542
  # Thanks to beaugunderson for the semantics
  RewriteEngine On
  RewriteCond %{QUERY_STRING} transport=websocket      [NC]
  RewriteRule /(.*) ws://localhost:9001/socket.io/$1 [P,L]
  ProxyPass http://localhost:9001/socket.io retry=0 timeout=30
  ProxyPassReverse http://localhost:9001/socket.io
</Location>

<Proxy *>
  Options FollowSymLinks MultiViews
  AllowOverride All
  Order allow,deny
  allow from all
</Proxy>
```

a2enconf コマンドを実行して ”etherpad-lite.conf” の設定を apache2.4 へ組み込み、apache2.4 を reload します。

```
# a2enconf etherpad-lite
Enabling conf etherpad-lite.
To activate the new configuration, you need to run:
  systemctl reload apache2

# systemctl reload apache2
```

これで以下の URL のように クライアントから etherpad-lite サーバへ 80 番ポートに接続して使えるようになりました。

<http://www.pcdennokan.wjg.jp/etherpad/p/debianmeeting20210220>

3.4.5 etherpad-lite の設定

デフォルトの設定ファイル ”settings.json” はクライアントからのアクセスに認証を必要としない設定になっています。最低限の認証を行えるように設定する例を紹介します。以下の設定は 1 つのアカウントを全ユーザで共用する設定になっています。

```

--- settings.json.template      2021-02-16 02:47:33.000000000 +0900
+++ settings.json              2021-02-18 22:15:43.144967658 +0900
@@ -315,7 +315,7 @@
 *
 * Note: "/admin" always requires authentication.
 */
- "requireAuthentication": false,
+ "requireAuthentication": true,

/*
 * Require authorization by a module, or a user with is_admin set, see below.
@@ -428,22 +428,26 @@
 *
 * follow the section "secure your installation" in README.md
 */

- /*
"users": {
  "admin": {
    // 1) "password" can be replaced with "hash" if you install ep_hash_auth
    // 2) please note that if password is null, the user will not be created
-   "password": "changeme1",
+   "password": "adminpassword",
    "is_admin": true
  },
-  "user": {
+  // "user": {
    // 1) "password" can be replaced with "hash" if you install ep_hash_auth
    // 2) please note that if password is null, the user will not be created
-   "password": "changeme1",
+   // "password": "changeme1",
+   // "is_admin": false
+   //}
+  "myuser": {
+    // 1) "password" can be replaced with "hash" if you install ep_hash_auth
+    // 2) please note that if password is null, the user will not be created
+    "password": "myuserpassword",
+    "is_admin": false
  }
},
- */

```

そのほか、<https://static.etherpad.org/index.html> で多数の etherpad-lite 用のプラグインがありますので実用するにあたり自分の利用方法にあるものを探してみるとよいと思います。

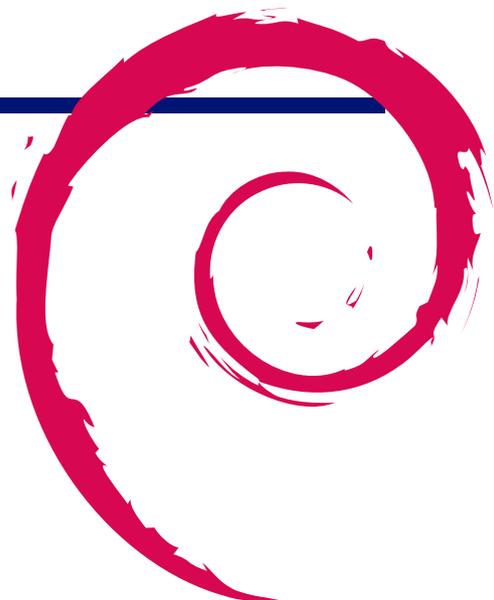
3.5 おわりに

etherpad-lite の歴史とインストール手順を説明し、etherpad-lite サーバを作ってみました。最近はクラウドサービスの Office 系アプリケーションでも同時編集が可能になっているサービスもありますが、オープンソースなアプリケーションを使って複数人でメモの同時編集する環境を構築できますのでオープンソースライフを楽しみつつ実用していただけるのではないかと思います。

3.6 参考情報

- etherpad.com
<https://etherpad.org/>
- EtherPad Open Source Release
<https://web.archive.org/web/20091221023828/http://etherpad.com/ep/blog/posts/etherpad-open-source>
- How to put Etherpad Lite behind a reverse Proxy
<https://github.com/ether/etherpad-lite/wiki/How-to-put-Etherpad-Lite-behind-a-reverse-Proxy>

4 メモ





Debian 勉強会資料

2021年2月20日 初版第1刷発行

東京エリア Debian 勉強会（編集・印刷・発行）
