

銀河系唯一のDebian専門誌

2019年10月19日

kFreeBSD入門



会
強
勉
\mathbf{h}
R
11

				トアップガイド 2019 年版	4
目次	•		3.1	はじめに	4
			3.2	Debian Ports & Debian	
1	最近の Debian 関連のミーテ			GNU/kFreeBSD	4
	ィング報告	2	3.3	Debian GNU/kFreeBSD σ	
1.1	2019 年 9 月度 東京エリア			インストール.......	6
	Debian 勉強会	2	3.4	Debian GNU/kFreeBSD σ	
2	事前課題	3		セットアップ.......	7
2.1	kenhys	3	3.5	Debian の開発ツールのセッ	
2.2	koedoyoshida	3		トアップ	11
2.3	, NOKUBI Takatsugu (knok)	3	3.6	その他の機能の紹介	12
2.4	$vv_v_ja_jp \dots$	3	3.7	おわりに	13
2.5	dictoss	3	3.8	参考文献	13
3	Debian GNU/kFreeBSD セッ		4	メモ	14

1 最近の Debian 関連のミーティング報告

1.1 2019 年 9 月度 東京エリア Debian 勉強会

2019 年 9 月 21 日 (土) に東京エリア Debian 勉強会を開催しました。会場は荒川区立町屋文化センターの会議室を 借りて行いました。参加者は 3 名でした。

杉本 典充

セミナーは行わず、各自で Hack タイムを行いました。Hack する中で日本語入力や日本語の処理に必要な辞書、 最近の機械学習業界の状況などの話題が出て意見を交わしました。

勉強会の後は参加者で懇親を深めました。

2 事前課題

今回の事前課題は以下です。

- 1. Debian Ports においてインストールしたことがあるアーキテクチャを教えてください (Which architecture of Debian Ports have you installed ?)
- 2. Debian Policy で理解を深めたいセクションがあれば教えてください(Please let me know if there are sections you want to understand in Debian Policy.)

2.1 kenhys

- 1. kfreebsd-amd64 / kfreebsd-i386
- 2. (回答なし)

2.2 koedoyoshida

- 1. hurd-i386, kfreebsd-amd64 / kfreebsd-i386
- 2. (回答なし)

2.3 NOKUBI Takatsugu (knok)

1. 一覧にはない何か

2. symbols 周り

杉本 典充

2.4 yy_y_ja_jp

- 1. kfreebsd-amd64 / kfreebsd-i386
- 2. (回答なし)

2.5 dictoss

- 1. kfreebsd-amd
64 / kfreebsd-i
386 $\,$
- $2.\,$ 8. Shared libraries, $\,9.\,$ The Operating System



3.1 はじめに

Debian Project では Debian GNU/Linux という Linux ディストリビューションを開発しており多くの開発者及 び利用者がいます。Debian Project では ユニバーサルなオペレーティングシステムを提供する考え方をもっており、 その理念に則り FreeBSD カーネルで動作する Debian GNU/kFreeBSD という別の OS^{*1} があります。

Debian GNU/kFreeBSD はその特異さゆえに Debian GNU/Linux と異なる点が多くあります。今記事では Debian GNU/kFreeBSD を触れるにあたり、どのようにセットアップを行うとよいか説明します。

3.2 Debian Ports & Debian GNU/kFreeBSD

3.2.1 Debian Ports とは

Debian Ports*2とは、さまざまな CPU やカーネルで動作するように移植を行うプロジェクトです。

FreeBSD カーネルで動作する Debian を作るプロジェクトがあり、その Debian のことを「Debian GNU/kFreeBSD」と呼んでいます ("k" は kernel のこと)。現在では Intel CPU のアーキテクチャのみあり ます (kfreebsd-amd64、kfreebsd-i386)。

3.2.2 Debian GNU/kFreeBSD のリリースの歴史

Debian GNU/kFreeBSD のリリースの歴史は以下となっています。

- 2011 年 2 月 6 日、Debian 6 (squeeze) おいてテクノロジープレビューとして stable 版をリリース。
 ftp.debian.org にてパッケージの配布を開始。
- 2013 年 5 月 4 日、Debian 7 (wheezy) において stable 版をリリース。
- 2014年9月、Debian 8 (jessie) でリリースするアーキテクチャを選定する過程で、kFreeBSD は開発や保守 に関わる人が少なく stable の維持と新規開発の両立ができていない状況が見られることを指摘される^{*3}。その 指摘を改善できなかったため、Debian 8 (jessie) では stable 版のリリースは見送られる^{*4}。
- Debian 9 (stretch) においても stable 版のリリースはされない状況が続く*5。
- 2018 年 5 月 31 日、Debian 7 (wheezy) の LTS が終了。これにて Debian GNU/kFreeBSD の stable 版は

 $^{^{*1}}$ Linux カーネルでなく FreeBSD カーネルでもない Debian GNU/Hurd というものもあります。

 $^{^{*2}}$ https://www.ports.debian.org/

^{*3} https://lists.debian.org/debian-devel-announce/2014/09/msg00002.html

^{*&}lt;sup>4</sup> 過去バージョンの利用者が困るであろうということで jessie-kfreebsd 版という裏バージョンが存在していた。

^{*&}lt;sup>5</sup> stretch-kfreebsd 版はリリースされませんでした。

サポートを終了。

- Debian 10 (buster) のリリースに向けて ftp.debian.org の整理を実施。
- 2019年5月25日、Debian GNU/kFreeBSD は Debian Ports(https://www.ports.debian.org/) へ移 転。これにより ftp.debian.org におけるパッケージの配布を終了。

3.2.3 Porterbox

Debian Project では多くのアーキテクチャを開発しているため、移植作業用にサーバを借りる「porterbox」という仕組みがあります*6*7。

bugreport にはあるアーキテクチャのみで発生する不具合も報告されますが、パッケージをメンテナンスしている 人がそのアーキテクチャの環境を持っていない場合があります。その場合は環境を自分で構築するか、porterbox を 利用してバグの修正作業を行います。

なお、Debian GNU/kFreeBSD では kfreebsd-amd64 の porterbox "lemon.debian.net" を運用しています*8*9。

3.2.4 Debian GNU/kFreeBSD 固有の Debian パッケージ

Debian GNU/kFreeBSD 固有のパッケージの例を紹介します。

kfreebsd-image パッケージ

Debian GNU/kFreeBSD の kernel イメージを収録したパッケージです。Debian Ports の kFreeBSD では kfreebsd-image-10.3 が利用できます。experimental には kfreebsd-image-11 があります^{*10}。

zfsutils パッケージ

zfsutils パッケージは ZFS を操作するコマンドを含んだパッケージです。インストール時のファイルシステムに ZFS を選択した場合はデフォルトでインストールされます。

freebsd-utils パッケージ

freebsd-utils パッケージは、FreeBSD 固有のコマンドを含んだパッケージです。/sbin/mount_*、/usr/sbin/jail などが入っています。

freebsd-net-tools パッケージ

freebsd-net-tools パッケージは、FreeBSD 向けのネットワーク操作のコマンドを含んだパッケージです。arp、 ifconfig、netstat、route コマンドなどが入っています。

freebsd-smbfs パッケージ

freebsd-smbfs パッケージは、Windows ファイル共有(SMB 共有)へアクセスするためのパッケージです。イン ストールすると、"/usr/sbin/mount_smbfs" コマンドが使えるようになります。

Windows ファイル共有先を mount するには以下のコマンドを実行します。

 $^{^{*6}}$ https://wiki.debian.org/PorterBoxHowToUse

^{*&}lt;sup>7</sup> 実際に借りた事例の報告「Debian の移植作業用のインフラを借りるには」https://tokyodebian-team.pages.debian.net/pdf2016/ debianmeetingresume201603.pdf

^{*8} https://db.debian.org/machines.cgi?host=lemon

^{*9 &}quot;New GNU/kFreeBSD Porterbox" https://lists.debian.org/debian-devel/2019/02/msg00313.html

^{*&}lt;sup>10</sup> FreeBSD の最新版は 2018 年 12 月 11 日にリリースした FreeBSD-12.0 です。

mount_smbfs -E UTF-8:CP932 -I {ファイルサーバの IP アドレス} -U {smb ユーザ名} //{ファイルサーバの IP アドレス}/{dir} {mount 先 dir

freebsd-ppp パッケージ

freebsd-ppp パッケージは、ダイアルアップする "/usr/sbin/ppp" コマンドを含んでいます。3G や LTE に対応 した USB モデムを使う場合に必要となります。

Debian GNU/Linux では ppp パッケージや wvdial パッケージでダイアルアップしますが、kfreebsd ではそれら は使えないため注意が必要です。

3.2.5 pf パッケージ

FreeBSD kernel がもつ Packet Filter と呼ばれるいわゆるファイアウォール機能を制御するコマンド "/sbin/pfctl" を含んだパッケージです。

/sbin/pfctlの設定ファイルは"/etc/pf.conf"であり、Linuxのiptables用設定ファイルと中身が全く異なります。

3.3 Debian GNU/kFreeBSD のインストール

3.3.1 インストールする PC

今回は ThinkPad X220 という Intel の Sandy Bridge 世代のノート PC にインストールをしてみます。このノート PC の有線 LAN と無線 LAN のチップは Intel 製であり FreeBSD カーネルの標準でドライバが提供されているため認識させやすいです。

3.3.2 インストールイメージの入手

開発者向けの Debian Installer は通常 https://www.debian.org/devel/debian-installer/ で配布してい ます。しかし、Debian GNU/kFreeBSD は Debian Ports へ移行したため、https://www.debian.org/devel/ debian-installer/ にある kFreeBSD 用インストーラは動作しなくなっています。

代わりに非公式版の Debian Installer の ISO イメージが用意されており、以下の ISO ファイルを利用してインストールします。

- http://jenkins.kfreebsd.eu/jenkins/view/cd/job/debian-cd_sid_kfreebsd-amd64/ws/build/ debian-unofficial-kfreebsd-amd64-NETINST-1.iso
- http://jenkins.kfreebsd.eu/jenkins/view/cd/job/debian-cd_sid_kfreebsd-i386/ws/build/ debian-unofficial-kfreebsd-i386-NETINST-1.iso

通常は kfreebsd-amd64 版を使うとよいでしょう*11

ISO ファイルを CD/DVD メディアへ焼いてインストールディスクを作成し、インストールディスクから PC を起動すると Debian Installer が起動します。

3.3.3 インストーラの表示言語

kfreebsd 版 Debian Installer は、日本語の表示ができません(インストーラでフレームバッファが有効になっていないと思われる)。そのため、LANG=C でインストールを進めます。

^{*&}lt;sup>11</sup> kfreebsd-i386 版を利用しても構わないのですが、ファイルシステムに ZFS を使う場合はメモリ不足になりがちなため注意してください。

3.3.4 パーティション構成とファイルシステム

Debian GNU/kFreeBSD を MBR 方式でブートする環境ヘインストールする場合は root パーティションを MBR の基本パーティションにする必要があります^{*12}(拡張パーティションにインストールすると grub のインストールに失敗します)。 kFreeBSD を占有する環境の場合は以下のパーティション構成でよいでしょう(以下では "/dev/ada0p1" のファイルシステムに UFS を選択しています)。

fdisk -1 /dev/ada0 Disk /dev/ada0: 238.5 GiB, 256060514304 bytes, 500118192 sectors Units: sectors of 1 * 512 = 512 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disklabel type: dos Disk identifier: 0x88f37f78 Size Id Type Boot End Sectors Device Start /dev/ada0p1 2048 273436671 273434624 130.4G a5 FreeBSD /dev/ada0p2 273436672 281249791 7813120 3.7G 82 Linux swap / Solari

ファイルシステムは通常の利用する分には UFS を選択することが多いです。他には ZFS というスナップショット 機能をもち多数のストレージを扱えるファイルシステムを選択することができます。

3.3.5 パッケージをダウンロードするミラーサーバの指定

Debian Installer が利用するミラーサーバは http://deb.debian.org/debian/ がデフォルトになっています。 しかし、kfreebsd-amd64 のパッケージは ftp.ports.debian.org にて配布しているためデフォルト値は利用できませ ん。ひとまずミラーは利用しない選択をして処理を進めて "Standard system utilities" のみをインストールしま す^{*13}。

3.4 Debian GNU/kFreeBSD のセットアップ

3.4.1 有線 LAN

有線 LAN は利用するドライバによってデバイス名が変化します (Intel の PC 向けの場合は"em0"、realtek の場 合は"re0")。

設定ファイルは Debian GNU/Linux と同じ "/etc/network/interfaces" を参照しますが、allow-hotplug 句は linux で使われる udev が提供している機能であることから kfreebsd では利用できず注意が必要です。

そのため、有線 LAN 接続環境がない状況で OS を起動すると有線 LAN による DHCP の IP アドレス取得がタ イムアウトするまで login プロンプトが出てこなくなります(起動に時間がかかる)。私はノート PC に Debian GNU/kFreeBSD をインストールする場合は以下コマンドを手動で実行してネットワークへ接続するようにしてい ます。

```
# vi /etc/network/interfaces
#auto em0 <-コメントにします
iface lan_home inet dhcp</pre>
```

ifup em0=lan_home

なお、ネットワークインタフェースの状態確認は昔ながらの"ifconfig"コマンドを利用してください(最近の linux と違い ip コマンドはありません)。

^{*&}lt;sup>12</sup> もし Debian GNU/Linux や Windows とデュアルブートしたい場合は、MBR 方式では基本パーティションの作成は 4 つまでという制 約を考慮したパーティション構成にする必要があります。

^{*&}lt;sup>13</sup> http://ftp.ports.debian.org/debian-ports/ を指定すればよさそうなものなのですが、入力してもパッケージが見つからないエ ラーが起こり、先に進めませんでした。

3.4.2 apt の設定

インストール時にミラーを指定しなかったため、/etc/apt/sources.list ファイルは中身がない状態になっています。 /etc/apt/sources.list は以下を設定します^{*14}。

```
# vi /etc/apt/sources.list
deb-src http://ftp.se.debian.org/debian/ sid main contrib non-free
deb http://ftp.ports.debian.org/debian-ports sid main
deb http://ftp.ports.debian.org/debian-ports unreleased main
deb-src http://ftp.ports.debian.org/debian-ports unreleased main
deb http://ftp.ports.debian.org/debian-ports experimental main
```

上記の apt の設定を行い apt-get コマンドを実行すると、 GPG キーエラーが発生して先に進めません。手動で keyring をインストールします。

wget http://ftp.jp.debian.org/debian/pool/main/d/debian-ports-archive-keyring/debian-ports-archive-keyring_2018.12.27_all.deb # dpkg -i debian-ports-archive-keyring_2018.12.27_all.deb

これで apt-get が動作するようになります。他の keyring も更新しておきましょう。

apt-get update
apt-get install debian-keyring debian-archive-keyring

3.4.3 locale の設定

Debian Installer では LANG=C を選択してインストールしているため、出力メッセージが英語になっています。 そのため locale を日本語に変更します (ただし、コンソール環境では日本語メッセージが化けるため C.UTF-8 の方 がデバッグしやすいかもしれません)。

dpkg-reconfigure locales
 -> ja_JP.UTF-8 を選択する

3.4.4 sshd のインストール

openssh-server パッケージを利用できます。以下コマンドでインストールできます。

apt-get install openssh-server

3.4.5 無線 LAN

無線 LAN のハードウェアの設定

ThinkPad X220 に搭載している無線 LAN チップは「Intel Centrino advanced-N 6205」です。

無線 LAN を利用するためにまずはメーカーが配布する無線 LAN 用 firmware をインストールします。ただ、/etc/apt/sources.list に non-free のバイナリパッケージのダウンロード先を指定をしていないため、パッケージ を直接ダウンロードしてインストールします。reboot して firmware を認識すると、dmesg に出力されていたファー ムウェアがロードできないというエラーがなくなります。

wget http://ftp.se.debian.org/debian/pool/non-free/f/firmware-nonfree/firmware-iwlwifi_20190717-2_all.deb
dpkg -i firmware-iwlwifi_20190717-2_all.deb
reboot

無線 LAN の通信を制御するデーモンである wpasupplicant パッケージ をインストールします。

^{*14} https://lists.debian.org/debian-bsd/2019/08/msg00005.html

無線 LAN のネットワーク設定

FreeBSD は物理無線 LAN インタフェースと論理無線 LAN インタフェースに分かれています。

ThinkPad X220 に搭載している Intel 製無線 LAN チップの場合は "iwn0" という物理無線 LAN インタフェース があります

があります。

#	# ifconfig iwn0
i	<pre>iwn0: flags=8802<broadcast,simplex,multicast> metric 0 mtu 2290 ether yy:yy:yy:yy:yy:yy media: IEEE 802.11 Wireless Ethernet autoselect (autoselect) status: no carrier nd6 options=23<performnud,accept_rtadv,auto_linklocal></performnud,accept_rtadv,auto_linklocal></broadcast,simplex,multicast></pre>
火⁻	下のコマンドで論理無線 LAN インタフェースを生成します。"wlan0"というインタフェースが生成されました。
# w	# ifconfig wlan create wlandev iwn0 wlan: Ethernet address: yy:yy:yy:yy:yy:yy

```
wlan0
# ifconfig wlan0
wlan0: flags=8802<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
    ether yy:yy:yy:yy:yy
    ssid " channel 1 (2412 MHz 11b)
    country US authmode OPEN privacy OFF txpower 14 bmiss 10 scanvalid 60
    bgscan bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 1 wme
    bintval 0
    media: IEEE 802.11 Wireless Ethernet autoselect (autoselect)
    status: no carrier
    nd6 options=23<PERFORMNUD,ACCEPT_RTADV,AUTO_LINKLOCAL>
```

接続する無線 LAN アクセスポイントの認証情報設定ファイルを作成します。

\$ wpa_passphrase apname1 appassword > wpa_apname1.conf
\$ cat wpa_apname1.conf
network={
 ssid=''apname1''
 #psk=''appassword''
 psk=e9fdcb43eba09b6342df30f14275625c8494e534799a82d6639b6124434ea627
}

無線 LAN アクセスポイントへ接続し、DHCP で IP アドレスを取得します。IP アドレスは論理インタフェースに

付与されます。

```
# wpa_supplicant -i wlan0 -c ./wpa_apname1.conf
Successfully initialized wpa_supplicant
ioct1[SIOCS80211, op=20, val=0, arg_len=7]: Invalid argument
wlan0: Trying to associate with zz:zz:zz:zz:zz: (SSID='apname1' freq=2427 MHz)
wlan0: Associated with zz:zz:zz:zz:zz
wlan0: WPA: Key negotiation completed with zz:zz:zz:zz:zz completed [id=0 id_str=]
wlan0: WPA: Group rekeying completed with zz:zz:zz:zz:zz:zz [GTK=CCMP]
```

```
# dhclient wlan0
# /sbin/ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
    ether yy:yy:yy:yy:yy
    inet6 fe80::a228:b4ff:fe18:b514%wlan0 prefixlen 64 scopeid 0x6
    inet 192.168.1.5 netmask 0xfffff00 broadcast 192.168.1.255
    ssid apname1 channel 4 (2427 MHz 11g) bssid zz:zz:zz:zz:zz:zz
    country US authmode WPA2/802.111 privacy 0N deftxkey UNDEF
    AES-CCM 3:128-bit txpower 15 bmiss 10 scanvalid 60 bgscan
    bgscanitvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
    wme roaming MANUAL
    media: IEEE 802.11 Wireless Ethernet 0FDM/54Mbps mode 11g
    status: associated
    nd6 options=23<PERFORMNUD,ACCEPT_RTADV,AUT0_LINKLOCAL>
```

無線 LAN 接続のトラブル

wpa_supplicant コマンドで無線 LAN アクセスポイントへ接続を試みたがエラーが発生し接続できない場合があり ます。その場合は以下を試すと接続できる場合があります。

- 接続先 SSID を 2.4GHz 帯のものに変更する。
- "# ifconfig wlan0 -ht40"を実行する。*¹⁵

3.4.6 音の再生

FreeBSD は OSS (Open Sound System) という仕組みで音声出力をします (ALSA は Linux 専用のサウンド システムです)。最近の PC には High Definition Audio 規格のチップが搭載されることが多いため、snd_hda.ko ド ライバでサウンドを出力することができます (snd_hda.ko は FreeBSD kernel に static link されています)。 以下のコマンドを実行すると、コンソール上で MP3 形式の音声ファイルを再生できます。

apt-get install mpg123 # mpg123 \${mp3 ファイルのパス}

3.4.7 電源関係

CPU クロックの制御は powerd パッケージの powerd が行っています。現在動作中の CPU クロック数は sysctl コマンドで取得できます。

電源に接続時は 2501 MHz で動作しています。

sysctl dev.cpu.0.freq
dev.cpu.0.freq: 2501

```
電源に接続しておらずバッテリー動作時は 800 MHz で動作しています。
```

sysctl dev.cpu.0.freq
dev.cpu.0.freq: 800

バッテリー残量を取得するには acpiconf コマンドを実行します。

```
# /usr/sbin/acpiconf -i 0
Design capacity:
Last full capacity:
                              57240 mWh
                              51530 mWh
Technology:
Design voltage:
                              secondary (rechargeable)
10800 mV
Capacity (warn):
Capacity (low):
                              2576 mWh
                              200 mWh
Low/warn granularity:
                              1 mWh
Warn/full
            granularity:
                              1 mWh
                              45N1172
Model number:
Serial number:
                                641
Type:
OEM info:
                              LION
                              SANYO
                              high
State:
Remaining capacity:
Remaining time:
                              99%
                              unknown
Present rate:
                              O mW
Present voltage:
                              12154 mV
```

サスペンドとハイバーネートについては未確認です。

3.4.8 KMS の有効化

KMS (kernel mode settings)を有効にするとコンソール画面の解像度が上がります。i915.ko が KMS 非対応で、i915kms.ko が KMS 対応のカーネルモジュールです。これらのカーネルモジュールを利用するときはどちらかひとつのみロードします。

^{*&}lt;sup>15</sup> デュアルチャネル接続を無効にして、20MHz 幅の電波で通信するように指示するコマンドです。筆者の自宅で利用している NEC 製 Wifi ルータ Aterm に接続するにはこれを指定しないとうまくつながりませんでした。

以下コマンドで KMS を有効にできます。

kldunload i915
kldload i915kms

再起動後も自動で kernel module をロードするには以下のように設定します。

vi /etc/modules
i915kms

3.4.9 X Window System

X Window System 用ビデオドライバ

ThinkPad X220 は Intel のオンボード GPU を搭載しています。そのため、ビデオドライバは xserver-xorg-videointel を利用します。インストーするには以下のコマンドを実行します。

apt-get install xserver-xorg-video-intel

今の kFreeBSD で安定動作させるには Intel のドライバの動作モードを古い UXA モードに変更するとよいようです。

```
# vi /etc/X11/xorg.conf.d/50-intel.conf
Section ''Device''
Identifier ''CardO''
Driver ''intel''
Option ''AccelMethod'', ''uxa''
EndSection
```

ウィンドウマネージャ

twm をインストールすることが可能です。それ以外の icewm、xfce4、lxde などはパッケージの依存関係が解決で きない状態のためインストールできません。グラフィカルログインマネージャの xdm、lightdm もパッケージの依存 関係が解決できない状態のためインストールできません。

ターミナルエミュレーターも必要になるため、xterm、eterm をインストールしておきます。

apt-get install twm xterm eterm

コンソール環境において root ユーザで startx コマンドを実行すると twm を起動できます。

startx

twmの画面上で左クリックをするとメニューが表示されますので、etermを起動すれば CLI の操作ができます*¹⁶。

3.5 Debian の開発ツールのセットアップ

3.5.1 インストールできるパッケージ

以下のパッケージは、/etc/apt/sources.list に設定したリポジトリからバイナリパッケージをインストールできます。

- make / bmake
- gcc-9 / g++-9
- clang-7
- dpkg-dev

^{*&}lt;sup>16</sup> xterm は "open ttydev: No such file or directory" というエラーが出て起動できませんでした。

- devscripts
- build-essential
- debhelper
- $\bullet~{\rm debootstrap}$
- \bullet subversion
- $\bullet~{\rm vim}$

3.5.2 インストールができない状態になっているパッケージ

以下のパッケージは ftp.ports.debian.org のあるパッケージ群で依存関係が壊れているためインストールできない パッケージです。

- clang-8
- git
- emacs / emacs-nox

3.6 その他の機能の紹介

3.6.1 コンテナ環境: Jail

FreeBSD には FreeBSD Jail というコンテナ型仮想化環境を実行する機能があります。freebsd-utils パッケージを インストールすることでコンテナを操作するコマンド群をインストールすることができます。コンテナ環境の作成は debootstrap、コンテナ環境の操作は jail、jls、jexec などが利用できます。

debootstrap --no-check-gpg sid ./jail_demo_1 http://ftp.ports.debian.org/debian-ports/

以下コマンドで chroot コマンドのように jail 環境に入れます^{*17}。

jail -c path=./jail_demo_1 command=/bin/bash

jail 環境を起動した後は、jls コマンドで jid を取得し、jexec で jail 環境の中で任意のコマンドを実行できます。

# jls JID IP Address Hostname 1 -	Path /root/jail/jail_1
<pre># jexec 1 cat /etc/debian_version</pre>	

3.6.2 Linux エミュレーション

debootstrap で Debian 6 squeeze の linux-i386 の コンテナ環境を作成しようとしたところエラーが発生しました^{*18}。

現在は動かないのかもしれません。

```
# debootstrap --no-check-gpg --arch=i386 squeeze ./linux_demo_1 http://archive.debian.org/debian/
ELF binary type "0" not known.
E: Unable to execute target architecture
```

3.6.3 完全仮想化環境

FreeBSD には OS を完全仮想化して動作させる virtualbox、bhyve がありますが、Debian GNU/kFreeBSD に はまだ移植されていません。

^{*&}lt;sup>17</sup> hostname を指定するオプション、ipv4 アドレスを指定するオプションは省略しています。

^{*&}lt;sup>18</sup> kfreebsd-image-10.3 の linux.ko は linux-2.6.32 の 32bit ABI のエミュレーションだったと記憶しています。

3.7 おわりに

Debian GNU/kFreeBSD のインストール方法とセットアップ方法について説明しました。bugreport で kfreebsd 環境で動かないと報告がある場合や kfreebsd を開発してみたい方は参考にしてみてください。

3.8 参考文献

- ^rDebian_GNU/kFreeBSD Debian Wiki _ https://wiki.debian.org/Debian_GNU/kFreeBSD
- "How to get a debian kfreebsd sid" (2019-08-25) https://lists.debian.org/debian-bsd/2019/08/msg00004.html
- 杉本典充 (2015). 「Debian GNU/kFreeBSD セットアップガイド 2015 年版」https://tokyodebian-team. pages.debian.net/pdf2015/debianmeetingresume201511-presentation-sugimoto.pdf

4 メモ



 Debian 勉強会資料

 2019年10月19日
 初版第1刷発行

 東京エリア Debian 勉強会 (編集・印刷・発行)