

月刊

Debian 専

日本唯一のDebian専門月刊誌

2010年12月18日

特集 1: CACert 入門

特集 2: 俺の `libsane` が火を噴くぜ!



1 Introduction

上川 純一

今月の Debian 勉強会へようこそ。これから Debian の世界にあしを踏み入れるという方も、すでにどっぷりとつかっているという方も、月に一回 Debian について語りませんか？

Debian 勉強会の目的は下記です。

- Debian Developer (開発者) の育成。
- 日本語での「開発に関する情報」を整理してまとめ、アップデートする。
- 場 の提供。
 - 普段ばらばらな場所にいる人々が face-to-

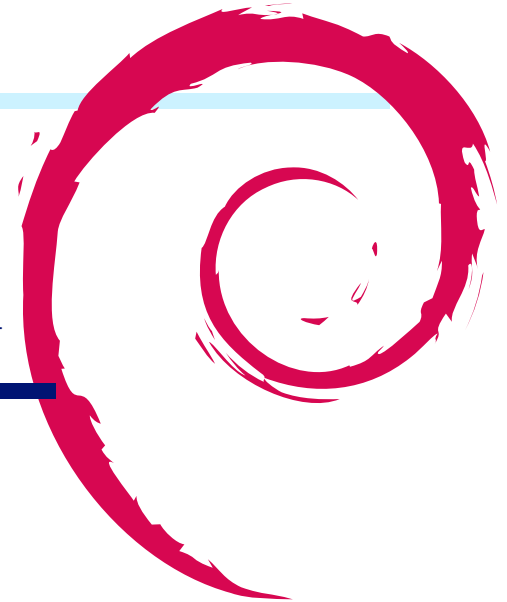
face で出会える場を提供する。

- Debian のためになることを語る場を提供する。
- Debian について語る場を提供する。

Debian の勉強会ということで究極的には参加者全員が Debian Package をがりがりとするスーパーハッカーになった姿を妄想しています。情報の共有・活用を通して Debian の今後の能動的な展開への土台として、「場」としての空間を提供するのが目的です。

会 勉 強 会 の ア ー キ ビ ト

目次		4	Debian Trivia Quiz	7
1	Introduction	1		
2	事前課題	3		
2.1	matsuu	3		
2.2	あらかやすひろ	3		
2.3	koedoyoshida	3		
2.4	キタハラ	3		
2.5	yamamoto	3		
2.6	本庄	3		
2.7	yyuu	4		
2.8	野島 貴英	4		
2.9	henrich	4		
2.10	岩松 信洋	4		
2.11	野首	4		
2.12	まえだこうへい	4		
2.13	上川純一	4		
3	最近の Debian 関連のミーテ ィング報告	6		
3.1	東京エリア Debian 勉強会 70 回目報告	6		
		5	2010 年の東京エリア Debian 勉強会をふりかえって	8
		5.1	基本的な数値	8
		6	CAcert の準備に何が必要か	11
		6.1	CAcert って何?	11
		6.2	CAcert でできること - 証明 書の作成、認証者としての活動	12
		6.3	認証を(して されて)みよう	13
		6.4	7年間の保管義務って?	15
		6.5	公式 CAcert イベントの開催 方法	15
		6.6	CAcert の主要資料と組織構成	18
		6.7	まとめ	21
		6.8	おまけ	21
		7	俺の libsane が火を噴くぜ!	22
		7.1	SANE の構造	22
		7.2	Code Flow	23
		7.3	スキャンするためのコード . .	23
		7.4	カラーのスキャン	24
		7.5	まとめ	25



2 事前課題

上川 純一

今回の事前課題は以下です:

- 「2010 年を振り返って自分が Debian でやったこと・Debian 界限でおきたこと。2011 年におきると予想すること、自分がやりたいこと。」

この課題に対して提出いただいた内容は以下です。

2.1 matsuu

やったこと・おきたこと

- Debian 勉強会に参加するようになった。
- VPS で Debian を使い始めた。
- Debian のパッケージを参考に Gentoo パッケージを作った。

予想・やりたいこと

- Debian と Gentoo は徐々に衰退していき、ALL YOUR DISTRIBUTION ARE BELONG TO CHROMEOS.

2.2 あらきやすひろ

予想

- 無事リリースされることによる Debian 派生ディストロの大崩壊と回帰。

やりたいこと

- Debian で仕事!

2.3 koedoyoshida

やったこと

- ddtss でレビュー (最近お休み中)
- 関西 KOF での関西 Debian ブースのお留守番
- 夏冬のイベントでの書籍の頒布

おきたこと

- Squeeze freeze

予想

- Squeeze release?

やりたいこと

- 予定は未定

2.4 キタハラ

やったこと

- 今年は一番何もしていない年ではないだろうか? ネットサーフィン用寝床 PC も死んだままだし。

やりたいこと

- 来年はこれを復活し、あと会社に Debian マシンを復活させたいですね。

2.5 yamamoto

やったこと

- なんかポチポチと勝手に ppc64 ポートを、マズいラーメン屋の頑固オヤジのごとく、細く長く続けていた。

おきたこと

- debian-ports に、sparc64 やら powerpcspe やら armhf やらがいきなり現れて、いっきに抜かれて行った。

予想

- 次は arm64 かな?

やりたいこと

- 頑固オヤジの迷惑なラーメンを、ひたすら生み出していく。
- arm64 ポートに参加できるといいな。

2.6 本庄

やったこと

- 温泉行きました。

やりたいこと

- また行きたいです。

2.7 yyuu

私事ではありますが、2009 年くらいから仕事で Debian を使えるようになりました。2010 年はその環境の整備に費やしているうちにあっという間に過ぎてしまいました。

やったこと

- lenny のシステムを 100 台以上の単位で扱うことができた
- 自分で個人的な apt のレポジトリを作って運用できるようになった

おきたこと

- 特に思いつきませんでした...

予想

- squeeze がリリースされる (2010 年?)

やりたいこと

- 一部の既存ホストを lenny から squeeze へ移行したい
- 新興プロジェクトの Debian パッケージ化にできるだけコミットしていきたい。今年 Apache Thrift などにパッチを提供できたが、今後はもう少し手を広げたい

2.8 野島 貴英

やったこと

- debian-sid の KVM 上で Opensolaris を稼働させた事。
- debian-sid の様々なソースコードいっぱい読んだこと。^{*1}

予想

- いわゆる携帯型 tabletPC 等へ debian-sid を搭載する Hack がそこいら中で起きる。

やりたいこと

- Contribution & Hack!Hack!Hack!(ソフトもハードも)

2.9 henrich

やったこと

- 5 月: NM プロセスを進めて DD になった
- 7 月: スイスから Debian 傘買ってみた :)
- 8 月: Debconf に参加した
- パッケージのアップデートを大体継続できた
- その他国内のイベントに参加した
- RC バグ潰しに参加した
- 多少ではあるが翻訳作業に参加した
- 手が付いていないことは以下
 - netbeans のパッケージ
 - eclipse-l10n のパッケージ (ソースを見つけれない...)
 - Knoppix-Math を Debian ベースに誘う
 - lenny インストール記事を JP ページに載せる

やりたいこと

- 開発者リファレンス訳の完了

- 初歩のプログラミングができるように (python あたり)
- debconf で何かしらの成果を出せるように事前準備
- l10n をもっと進められるようにもっとステップと成果を明確にしたい

2.10 岩松 信洋

やったこと

- 組み込み関係のサポート。
- SH4 の emdebian サポート。
- Macbook 関係のサポート。
- DD になってからの初めてのリリース作業。
- Debconf への参加

おきたこと

- Non-packaging contributors
- backports が正式サービスになった。
- Squeeze フリーズ
- Miniconf が多く行われた。
- fjp の急逝

予想

- Debian では SH4 unstable 入り
- 4G ネットワークがじわじわと浸透
- Android を使ったタブレットデバイス祭り

やりたいこと

- web 系の開発
- 関数型言語の習得

2.11 野首

2010 年はほとんど何もしませんでした... さすがに squeeze は 2011 年にはリリースされていると思いたい。

2.12 まえだこうへい

やったこと

- Debian 勉強会の運営 (いつもどおり)
- 山田さんの JP への勧誘・加入 (よくやった)
- 他の勉強会での勧誘 (Web アプリ開発者に交友関係は増えたが Debian への直接的な成果はなし)

おきたこと (予想?)

- Squeeze には CouchDB 1.0.1 が入らなさそう。

やりたいこと

- いろいろ pending なものを再開する。

2.13 上川純一

やったこと

- 子供が生まれた、Debian 活動レベルが低下した。
- 携帯電話は Android 携帯電話がメインになって、パソコンをあまり起動しなくなった。
- 勉強会にくるようなメンバーで、Smart Phone (iPhone or Android) をもっていない人に会わなく

^{*1} <http://d.hatena.ne.jp/nozzy123nozzy/>

なってきた。
予想

- さらなるスマートフォンとタブレットの普及とそれに伴うより自由で便利な環境への渴望。
- Debian としてはその環境との相互運用性の向上と、そ

のデバイス自体で動くシステムとしてのすすみかたが
二つ考えられる。
やりたいこと

- 相互運用性は少なくとも向上したいと考えている。

3 最近の Debian 関連のミーティング報告

上川純一



3.1 東京エリア Debian 勉強会 70 回目報告

東京エリア Debian 勉強会。参加者は、やまださん、前田さん、上川、野島さん、服部さん、キタハラさん、emasaka さん、matohara さん、鈴木さん、吉田@板橋さん、高橋齊大さん、荒木 (靖) さん、本庄さん、やまもとさん、吉野さんの 15 人でした。

ファイルシステムについてかたりました

まず最初に上川が ext4 について紹介しました。

nilfs について紹介しました。

btrfs について紹介しました。

ceph について紹介しました。

4 Debian Trivia Quiz

上川 純一

ところで、みなさん Debian 関連の話題においついていますか？ Debian 関連の話題はメーリングリストをよんでいると追跡できます。ただよんでいるだけでははりあいがないので、理解度のテストをします。特に一人だけでは意味がわからないところもあるかも知れません。みんなで一緒に読んでみましょう。

今回の出題範囲は `debian-devel-announce@lists.debian.org` や `debian-devel@lists.debian.org` に投稿された内容と Debian Project News からです。

問題 1. DACA ってなんですか？

- A Debian Admin Coaching Association
- B Debian を使うと (A) あの子と (C) クリスマスに (A) アレができるかもしれない
- C Debian's Automated Code Analysi project

問題 2. DebConf11 はいつ開催されるでしょう

- A 2011/6/24 - 30
- B 2011/7/24 - 30
- C 2011/8/24 - 30

問題 3. squeeze の Linux カーネルは一味違います。何が違うでしょう。

- A non-free firmware を排除した
- B Tux くんを排除した
- C 一つのカーネルイメージで kFreeBSD と Linux を提供します

問題 4. 2010/12/17 の時点で RC ハグはいくつあるでしょう。

- A 3
- B 83
- C 183

問題 5. New Maintianer フロントデスクに追加されたメンバーは？

- A Xavier Oswald
- B Enrico Zini
- C Kenshi Muto



5 2010 年の東京エリア Debian 勉強会をふりかえって

上川 純一

今月で 6 年目の Debian 勉強会が終了しました。Debian Developer の数も着々と増えていき、当初の目標が達成されてきていますね。

今年東京エリア Debian 勉強会常連のやまねさんが Debian Developer になりました。苦節 6 年おめでとうございます。

Debian Maintainer 申請も複数通りました。

5.1 基本的な数値

Debian 勉強会は毎回事前課題事後課題を設定しており、予習復習を必要だとうたっている勉強会です。実際にどれくらいの人が出席しているのか、またその人たちがどれくらい事前課題・事後課題を提出しているのか、確認してみましょう。図 1 です。値は一年の移動平均です。

結果を見ると事前課題の提出率が 2009 年に低下傾向にあったのが、2010 年に反転上昇しています。これは Debian 勉強会予約システムの導入により事前課題の提出が簡単になったことと関係ありそうです。また、別の傾向として事後課題 (ブログ) の率が低下傾向にあります。もうブログは流行らないのかもしれません。

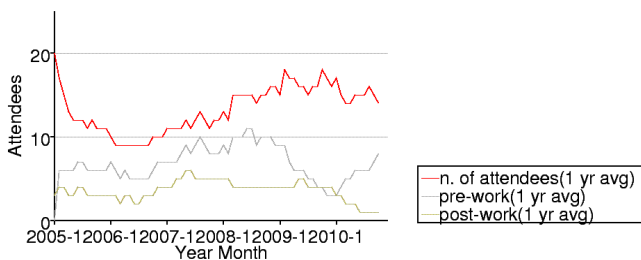


図 1 東京エリア Debian 勉強会事前課題・事後課題提出実績 (12ヶ月移動平均)

毎回の参加者の人数と、その際のトピックを見てみます。今年はとくに目立った感じの回はないですが、参加者についての記録が散逸しているようです。

今年の会場ですが、東京大学、筑波大学、木更津高専とアカデミックな場所の利用が増えました。^{*2}。また、NIFTY さんの会場も使い、公営の会議室 (杉並区、オリンピックセンターなど) は数回しか利用していません。

^{*2} OSC も明星大学で開催されました

表 1 東京エリア Debian 勉強会参加人数 (2005-2006 年)

	参加人数	内容
2005 年 1 月	21	秘密
2005 年 2 月	10	debhelper 1
2005 年 3 月	8	(早朝) debhelper 2、social contract
2005 年 4 月	6	debhelper 3
2005 年 5 月	8	DFSG、dpkg-cross、lintian/linda
2005 年 6 月	12	alternatives、d-i
2005 年 7 月	12	toolchain、dpatch
2005 年 8 月	7	Debconf 参加報告、ITP からアップロードまで
2005 年 9 月	14	debconf
2005 年 10 月	9	apt-listbugs、バグレポート、debconf 翻訳、debbugs
2005 年 11 月	8	DWN 翻訳フロー、statoverride
2005 年 12 月	8	忘年会
2006 年 1 月	8	policy、Debian 勉強会でやりたいこと
2006 年 2 月	7	policy、multimedia
2006 年 3 月	30	OSC: debian 勉強会、sid
2006 年 4 月	15	policy、L ^A T _E X
2006 年 5 月	6	mexico
2006 年 6 月	16	debconf、cowdancer
2006 年 7 月	40	OSC-Do: MacBook Debian
2006 年 8 月	17	13 執念
2006 年 9 月	12	翻訳、Debian-specific、oprofile
2006 年 10 月	23	network、i18n 会議、Flash、apt
2006 年 11 月	20	関西開催: bug、sid、packaging
2006 年 12 月	14	忘年会

表 2 東京エリア Debian 勉強会参加人数 (2007-2008 年)

	参加人数	内容
2007 年 1 月	15	一年を企画する
2007 年 2 月	13	dbcs、dpatch
2007 年 3 月	80	OSC 仮想化
2007 年 4 月	19	quilt、darcis、git
2007 年 5 月	23	etch、pbuilder、superh
2007 年 6 月	4	エジンバラ開催: Debconf7 実況中継
2007 年 7 月	18	Debconf7 参加報告
2007 年 8 月	25	cdn.debian.or.jp
2007 年 9 月	14	exim
2007 年 10 月	30	OSC Tokyo/Fall(CUPS)
2007 年 11 月	19	live-helper、tomoyo linux kernel patch、server
2007 年 12 月	11	忘年会
2008 年 1 月	23	一年を企画する
2008 年 2/29,3/1	36	OSC
2008 年 3 月	37	データだけのパッケージ、ライセンス
2008 年 4 月	17	バイナリパッケージ
2008 年 5 月	20	複数のバイナリパッケージ
2008 年 6 月	10	debhelper
2008 年 7 月	17	Linux kernel patch / module パッケージ
2008 年 8 月	10	Debconf IRC 会議と Debian 温泉
2008 年 9 月	17	po4a、「Debian メンテナのお仕事」
2008 年 10 月	11?	OSC Tokyo/Fall
2008 年 11 月	17	「その場で勉強会資料を作成しちゃえ」Debian を使った L ^A T _E X 原稿作成合宿
2008 年 12 月	12	忘年会

表 3 東京エリア Debian 勉強会参加人数 (2009-2010 年)

	参加人数	内容
2009 年 1 月	12	一年を企画する
2009 年 2 月	30	OSC パッケージハンズ オン
2009 年 3 月	23	Common Lisp, パッ ケージ作成
2009 年 4 月	15	Java Policy, ocaml, 開 発ワークフロー
2009 年 5 月	13	MC-MPI パッケージ化、 Erlang、Android アプ リ、DDTP
2009 年 6 月	14	DDTP・DDTSS、 bsdstats パッケージ、 Debian kFreeBSD
2009 年 7 月	4	スペインにて Debconf 9
2009 年 8 月	14	スペイン Debconf 9 参 加報告
2009 年 9 月	26	GPG キーサインパーテ ィー
2009 年 10 月	30	OSC Tokyo Fall
2009 年 11 月	12	Octave, R, gnuplot, auto-builder
2009 年 12 月	10	忘年会
2010 年 1 月	17	東京大学にて新年会
2010 年 2 月	11	Debian 温泉,ocaml,haskell
2010 年 3 月	12	weka,fftw,dpkg v3 quilt
2010 年 4 月	15	upstart,piuparts,debtags
2010 年 5 月	22	筑波大学,kernel
2010 年 6 月	12	OSC-Do リハーサル
2010 年 7 月	0	キャンセル
2010 年 8 月	3	Debconf (NYC)
2010 年 9 月	30	OSC Tokyo/Fall
2010 年 10 月	13	俺の Debian な一日
2010 年 11 月	15	ext4,btrfs,nilfs,ceph
2010 年 12 月	14	cacert, libsane



6 CAcert の準備に何が必要か

山田 泰資

6.1 CAcertって何？

CAcert プロジェクトとは誰もが低コストに証明書を持てるようにすることで、安全な環境・通信を広く実現しようというプロジェクトです。2002 年に発足し、2003 年には非営利法人の CAcert Inc. がオーストラリアで設立され、以来 7 万人のメンバーに対して 18 万件の証明書（クライアント認証用、サーバ認証用、S/MIME 署名用、コード署名用、etc）を発行してきています。

通常、個人や組織内で自己発行した証明書は「オレオレ証明書」などと呼ばれてあくまで閉じた世界での利用になります。しかし、このプロジェクトでは CAcert がルート証明機関として、そのルート証明書が各種 OS やブラウザに組み込まれ、VeriSign 社などによる商用サービスに近い水準で世界的に信頼される状態を目指しています。このため、コミュニティベースのプロジェクトとは言っても、厳格かつ高い水準の認証基準や運営ポリシーが定められています*3。

この高い水準の認証局運営を低コストで行うため、CAcert は次の 2 つを運営の柱としています：

1. 「信頼関係の構築・維持」と「証明書発行」の分離
2. 「信頼度・経験値のポイント化」と、「ポイントによる発行制限」

まずは前者から説明します。通常、証明書の発行では申請者の身元確認などを法的文書に基づいて行い、その上で証明書を発行します。この前者が高コストで、

1. 提出された申請内容が真正かどうかの確認
2. 後日の紛争に備えての関係書類の（法的に有効な）紙での保管

などで事務コストがかかります。一方、証明書の発行は秘密鍵の安全な保管などセキュリティ面のコストは必要なものの、比較的安価です。そこで、CAcert では

1. 本人確認と関係書類の保管はコミュニティベースで各自が実施
2. それに基づいた証明書の発行は CAcert がルート CA となって実施

という認証と発行の分離を行っています。コミュニティ側で行った認証結果をポイントという形で CAcert に登録し、CAcert はポイントに応じて各種の証明書を発行する訳です。

そしてこのポイント制度も CAcert の特色の 1 つです。CAcert Inc. 自身は本人確認を行っていないので、認証をする側もされる側もコミュニティメンバーという事になります。この仕組みの中で十分な認証を行うため、

1. 認証「される」人が申請できる証明書の形式・機能はポイントに応じる

*3 実は途中沈滞していたものの、ここ数年で急速に整備されてきました

2. 認証「できる」ようになるのは 100pt に到達してからで、更に試験合格が必要
3. 認証時に付与できるポイントも 100pt の成り立てでは少なく、経験に応じて増える

と 1 回の認証ではなく多人数間で認証を行う (Web of Trust と呼びます) 方式を導入しています。

6.2 CAcert でできること - 証明書の作成、認証者としての活動

さて、まずは一番利用するであろう証明書の作成について紹介します。実際のところ、認証を受け、利用するのみのメンバとしてであれば、これが CAcert の唯一の機能です。

これは <https://cacert.org/> に行き、メンバ登録を行うことで作成できます。ただし、認証可能メンバ (CAcert Assurer) による認証・ポイント付与を受けていない場合、作成できるのは最低限の機能を持つ証明書のみとなります。以下が保有ポイントによる作成可能な証明書などメンバ権限の一覧です：

保有ポイント	ステータス	できること
≥ 0	Member	実名抜き (メールアドレスのみ入る) のサーバ・クライアント用証明書の発行 (半年有効)
≥ 1	Member	上と同様だが、署名元のルート証明書が Member Root 証明書に格上げされる
≥ 50	Assured Member	実名入りのサーバ・クライアント用証明書の発行 (2年有効)
≥ 100	Prospective Assurer	上に加え、コードサイン証明書発行 (1年有効)
≥ 100 + 試験通過	Assurer	上に加え、オンライン試験に合格した場合、認証可能メンバとして他者を認証し、最大 10 ポイントまで付与できる。また、自身は以後の認証の度に 0-2EP を受領する
≥ 100 + 10EP + 試験通過	Assurer	上と同様だが、付与可能ポイント上限が 15 ポイントとなる
≥ 100 + 20EP + 試験通過	Assurer	上と同様だが、付与可能ポイント上限が 20 ポイントとなる
≥ 100 + 30EP + 試験通過	Assurer	上と同様だが、付与可能ポイント上限が 25 ポイントとなる
≥ 100 + 40EP + 試験通過	Assurer	上と同様だが、付与可能ポイント上限が 30 ポイントとなる
≥ 100 + 50EP + 試験通過	Assurer	上と同様だが、付与可能ポイント上限が 35 ポイントとなる。また、コミュニティ内において Experienced Assurer / Senior Assurer としての役割に進む入口に達したと見なされる

表 4 ポイントによるアカウントステータスと権限の一覧

上の通り、証明書の発行を受ける利用者としては 100pt が上限となります。これ以上はどれだけ認証を受けても Assurance Point は増えません (システム的に増えたように見えても無効)。しかし、一方で試験を受け、Assurer として他者を認証すると、その度に最大 2EP が与えられます。表中の 10EP/20EP/... とある部分がそれで、これを Experience Point と呼びます。以後はこちらを蓄積することで、更に上位のステータスを得る形になります。

ちなみに試験は選択式 (25 問中 20 問正解で合格) で、実はそんなに難しくありません。100pt に達する前から受けることができ、合格するまで何度やっても構いません (トレーニング的な意味で定期再受験が推奨されている)。合格しておけば 100pt になった瞬間から Assurer になれるので、皆さん受けておいてはいかがでしょうか?*4以下が試験画面です：

*4 <https://cats.cacert.org/> から受けられますが、クライアント証明書が必要です



図2 オンライン試験（選択式）の画面（全25問）

最後に、さらに上位の Assurer として Experienced / Senior Assurer というものがあります。これらはイベント内での認証会の開催では Experienced Assurer が最低3名必要であったり、コミュニティ内での調停やトレーナーには Senior Assurer が指名されるなど、より幅広い関わりを持つ上で必須の重要な立場となります。Experienced Assurer は 50EP 以上（つまり、100pt 分の認証を受けた後、最低 25 人以上を認証した状態）が条件ですが、Senior Assurer の認定には

1. Experienced Assurer であること
2. さらに、co-auditor^{*5}の試問を受け、合格すること
3. さらに、各国^{*6}で開催される ATE(Assurer Training Event) に参加し、トレーニングを受けていること
4. さらに、宣誓 (CARS - CAcert Assurer Reliable Statement) を行うこと

という高いハードルがあります。組織としての CAcert が一通り機能するには Senior Assurer の存在がキーとなるため、日本で CAcert が完全に立ち上がったと言えるのは、この Senior Assurer が生まれた時になるでしょう。

6.3 認証を（して | されて）みよう

最後までハードルは高いのですが、まずは第一歩からということで、1対1で認証を（する | される）時の手順を紹介します。

まずは双方で以下を用意します（標準的な手順の場合）:

^{*5} 認証ポリシーの策定や監査を統括する Assurance Office の委託を受けて活動している経験豊富な Experienced Assurer

^{*6} 現時点ではほぼ欧州でのみ開催 …

認証する側 (Assurer) が用意するもの

1. 相手に名乗るための名刺 (ただし、口頭でもよい^a)
2. 白紙・未署名の CAP(CAcert Assurance Program) Form (相手が忘れた時用)^b
3. 紫外灯などの、証明書の偽造検出のための機材^c
4. 法的責任や義務を説明するための CCA(CAcert Community Agreement) (できれば渡せるよう紙で)

^a 不特定多数のイベントでは、0点付与/調停となる可能性もあるので、口頭に留めるのが正解？

^b <https://secure.cacert.org/cap.php>

^c 日本政府発行のパスポートなどをこれで確認する

認証される側 (Member) が用意するもの

1. CAcert に登録したメールアドレス (複数登録の場合はプライマリのもの)
2. 身元証明書。2つ以上が望ましく、最低1つは政府発行かつ誕生日入りである必要がある。また、最低1つは写真入りである必要がある
3. 未署名の CAP Form (署名以外は事前記入・印刷 OK)^a

^a <https://secure.cacert.org/cap.php> - なお、ログインすれば事前記入済みのものをメニューから選択・印刷もできる

中央集権型の CAcert では、証明書の階層構造の安全を守るためと、認証手続きが実際に法的責任が生じる契約であることから、GPG キーサイン会よりも高い基準での偽造検出や確認を要請しています。上の紫外灯での真正性確認と、未署名書類へ目の前で署名することの確認義務などがそれにあたります。

さて、続きです。お互いの準備が整ったら、以下の手順で説明・確認・署名を行います。

!! 手順は説明・確認・署名の順番です !!

大事なことなので二回書きました。

1. まず、Assurer から CCA の概要、特に法的に発生する義務と責任について明確に説明して下さい。骨子としては以下の3点ですが、Assurer は下記の背景や CAcert の意義を合わせて説明する義務があります。
 - (a) 係争となった場合、各国内の裁判所に提訴する権利は放棄し、CAcert が Assurer の中から任命する調停者の裁定に従う義務があること
 - (b) 誤用や誤認証を行い損害が発生した場合、補償責任があること。ただし、最高でも 1000EUR (2010/12 の相場で約 10 万円強) となる
 - (c) 証明書の運用・保証範囲やプライバシー保護などに関して、公式文書群 (COD - CAcert Official Document) の規定に従うこと
2. Member が上記の義務・責任に同意するなら次に進みます
3. 次に、Assurer は Member に CAP Form への記入と署名を求め、併せて証明書を受け取って下さい。なお、署名と提出は目の前で行われる必要があります (事前署名は不可)
4. Assurer は内容を確認して下さい。確認ポイントは以下の通りです：
 - (a) 責任・義務を規定する CCA に同意する旨が明記・チェックされていること
 - (b) 氏名・誕生日が一致し、また、写真からも当人であることが確認できること
 - (c) メールアドレスが [cacert.org](https://secure.cacert.org) に確かに登録されていること
 - (d) 書類への署名が証明書に記載の署名 (あれば) と同じであること
 - (e) 証明書が失効しておらず、真正と見えること (紫外灯チェック、写真の上にハンコがかかっているか、等)
 - (f) 相手が氏名・誕生日など、証明書記載事項の質問に対して正答すること
5. 不備があるなどで確信が持てない場合、以下の対応を行います：
 - (a) 仕組み上、付与ポイントが少なくなったり、調停が必要となる可能性があることを伝えて下さい
 - (b) 表記違いがある場合、証明書の記載内容が優先されます。CAcert 登録内容の修正について当人と調整し、

dispute (調停依頼^{*7}) を Assurer より提起して下さい。ポイント付与は修正後に行います。

(c) 複数の証明書で氏名表記に揺れがあるものの同一人物と判断できる場合は、CAP Form にすべて列挙の上で認証して下さい。

(d) 芸名であっても、それが規定の証明書で証明可能であれば、認証して構いません

(e) 称号 (PhD 等) は使わないで下さい。証明書の記載内容以上のものを認証してはなりません

6. 以上で両者の対面での手続きは終了です。ここで別れて下さい。

7. 最後に、CAP Form の裏面にメモを残します。困った点、書類が古かった、妙に急かされた等の引っかかる点があったらそれを、また、面白い用法や、相手が運営参加希望者だった場合はその得意分野などです。後日の連絡や調停時に使います。

以上が当日のやりとりとなります… が、Assurer の仕事はまだ終わっていません！認証パーティが終わったら、Assurer の手元には署名入り CAP Form が沢山残るはずですよ。これらを元に、以下の事後処理を行います：

1. CAP Form に対して認証者として署名する

2. cacert.org より、CAP Form 記載のメールアドレスに対してポイントを付与する。付与するポイント数は、裏面メモなどを見ての確信度に応じて0点から最高点まで調整

3. 処理を終えた CAP Form を、7年間保管できる場所にしまう ← ← ← 説明後述

さて、ここで「0点を付与」と「手続き自体をしない」は意味が異なります。「0点」は「相手を信頼できなかった」ではなく

書類が未知すぎて、点数を付けたくてもできませんでしたー！mOm

という意味になります。0点を貰った人は、誤解ないようにしましょう。一方、手続きをしないというのは

いや、この(人 | 書類)はおかしいだろ JK

という、どう見ても本物ではなかったなどの疑念が大きい場合です。この場合は裏面メモを取っておき、ポイント付与ではなく調停依頼を support@cacert.org に要請することも考えましょう。

6.4 7年間の保管義務って？

おそらく GPG キーサインとの最大の違いは、この点になります。

単に認証基準が高いというだけでなく、CAcert は

デジタル空間の証明階層を実世界の法的文書でバックアップする

という意義を持っています。このため、紙ベースで署名・作成された CAP Form は Assurer に保管義務があるだけでなく、その電子化も禁止されています^{*8}。

もし自分では保管し続けることが困難になった場合は、support@cacert.org に調停依頼を要請し、指示を仰いでください。

6.5 公式 CAcert イベントの開催方法

1対1での認証と違い、正式な認証会の開催となると多くのものが必要となります。基本要件は先の1対1と同じなのですが、公式 CAcert 認定イベントとして開催する場合、各種の規定があります。要件としてあるのか、単に「円滑な開催、信頼性の高い認証」のために推奨されているものか判然としない部分もありますが、マクロレベルで

^{*7} この単語だけだと紛争・係争ですが、このような修正依頼なども含まれるため、調停を用語としました

^{*8} CAP Form 毎に署名を付加して各自で保管とか考えましたが、再びデジタル空間に基盤が戻ってしまうので不可のようです...

の開催の流れは以下ようになります。

1. Event Office へ開催提案を送る (<http://wiki.cacert.org/Officers>)
2. CAcert との窓口になる主催者 (EventOffier) を協議して決める
3. 開催日・場所を決め、スポンサーを付ける場合は予算案を詰める^{*9}
4. 計画策定。出展内容や必要資材・搬入の調整、更には当日に Assurer として参加を求める場合は事前の Assurer 人員の教育や宿泊の手配を行う
5. 事前準備。ブースの設営、配布物の用意、買い出しなど
6. イベント実施
7. 後片付け。撤収の他、再利用可能なバナーなどの資材は回収・整理する
8. 完了報告。EventOfficer よりレポートを Event Office に送って終了

これはあくまで CAcert 公式イベントとする場合なので、そうでない場合は1対1の認証を基本として、この章の内容は準備内容の参考に使って下さい。

6.5.1 準備チェックリスト

ここでは推奨されている確保すべき人員・資材の一覧を列挙しておきます。これを満たさなくては正規の認証手順を経たと認定されないということはなさそうですが、後から不備・無効となるのを避けるため、慣れるまでは心配な点は CAcert 側の Event Office に相談するとよいでしょう。

役割	人数	注記
Event Organiser	1	主催者兼進行役。一番の経験者がよい。Event Office と相談して決める
Experienced Assurer	3+	認証担当。他ブースを訪問して証明して回る巡回証明役もいるとよい
Assurer	2+	EA の補助として CAP Form の事前確認を行い、EA をフル回転させる
ガイド役	2+	行列の整理や質疑、派生して起こる問題の対処など

表 5 CAcert イベント開催に必要な人員

資材	個数	注記
紫外灯	1+	並行処理する認証手続きの個数だけ用意
机	1+	並行処理する認証手続きの個数だけ用意
椅子	1+	並行処理する認証手続きの個数だけ用意
ネット回線	0+	事前印刷が十分できていれば必須ではないが推奨
PC	1+	資料閲覧、ポイント付与作業、新規登録用 (Knoppix 推奨)
プリンタ	1	CAP Form 不足や配布資料の現場量産用
ペン	10+	参加人数や進行方法に基づいて本数は考える
募金箱	1	経費のカバー用 (事前告知があれば、正式に徴収してもよい)

表 6 CAcert イベント開催に必要な資材

^{*9} 本気で公式イベント化する場合、どうも色々支援がある？

資料	個数	注記
Assurer 連絡名簿	1	認証に関する疑義対応や後日のフォローアップ用
CAP Form (1p, 記入済)	30+	未署名。開催日、場所、Assurer 名まで記入
CAP Form (1p, 白紙)	20+	新規 Assurer に手伝ってもらう用
CCA (4p)	50+	Assurance に伴う契約義務・責任の解説用
CCA (巨大張り出し用)	1	壁に張り出して使う
Assurance Policy (8p)	1+	質疑回答、参照用
Assurance Handbook (29p)	1+	質疑回答、参照用
Root Distribution License (1p)	1	質疑回答、参照用
Dispute Resolution Policy (6p)	1	質疑回答、参照用
Practice On Names (4p)	1+	各 Assurer に 1部。表記揺れへの対応ガイド
Practice On ID Checking (4p)	1+	各 Assurer に 1部。証明書内容の検査ガイド
PoJAM (4p)	1	未成年への認証、未成年による認証に関する規定
CPS(28p)	1	発行証明書の内容・用法の規定
その他	0+	CAcert Office と協力する場合、IR 資料や景品が多数用意される

表 7 CAcert イベント開催に必要な資料

多人数を相手にあまり時間を取らずに説明・認証をするため、CCA(CAcert Community Agreement)の印刷・配布や書類不備の防止のための確認員の準備が成功のキモになるようです。日本で開催する場合は、説明会の形式を取って、各座席に一部ずつ置いておき、解説の進行と共に記入して貰うような形がよいのではないのでしょうか。

6.5.2 当日の進行手順(案)

具体的な進行方法については特に案内が見当たらなかったため、開催未経験者ですが進行案をここに書き出しておきます。

1. まず、クラスルーム形式で CAcert および発生する義務・責任の説明を行う。入室時に各自に 3-5 枚^{*10}CAP Form を渡し、説明中に署名以外は全部記入してもらう。
2. 次に、Assurer と Member(予定者) で横並びに列を組んで、以下の図の通り Member にはカニ歩きをしてもらいながら認証をする。
3. 認証で新たな Assurer が生まれたら、Experience Point 獲得のためできるだけ Assurance 側に回ってもらう。

^{*10} その場にいる Assurer のランクや人数による

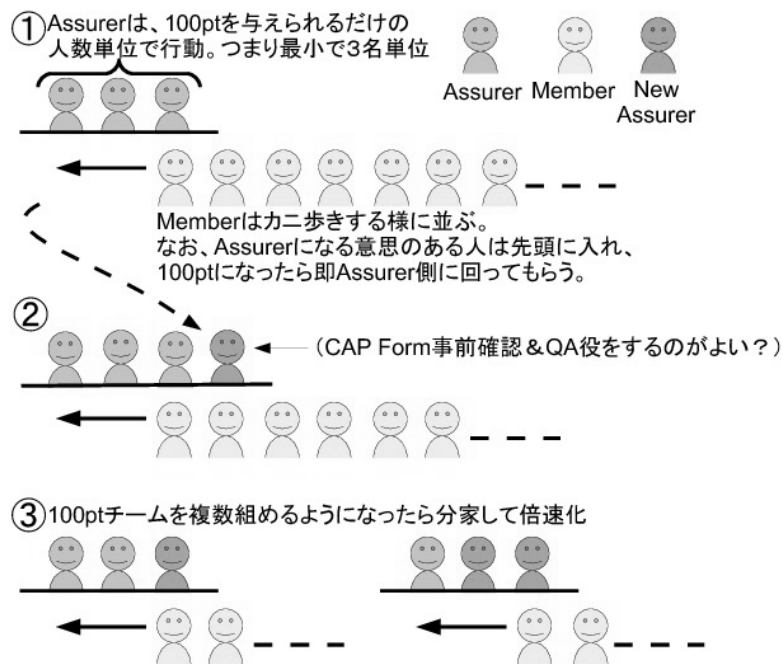


図3 CACert サイン会の進行案

以上の進行で Assurer の数を早く増やすことができます。

6.6 CACert の主要資料と組織構成

CACert Wiki などを読んでも資料が大量にあって大変、しかも組織構成も訳がわからなかった、という方もいるのではないかと思います（自分のこと）。これは法的に監査を受け、商用の証明局並みの信頼度を獲得して普及を図るというゴールのために整備を重ねてようになってきたのですが、理解しやすくなるように CACert の主要ポリシ・資料の関係図と位置付け、また組織構成をまとめてみました。

まずは主要文書の一覧から。なお、公式文書（組織内の文書として法的に有効なもの）はすべて COD (CACert Official Document) というコード番号が振られています。

文書番号	タイトル (略称)	状態	内容
COD1	Policy on Policy (PoP)	POLICY	ポリシー策定に置ける IETF 風のコンセンサスベースの運営方針およびドキュメント状態の管理方法を定める
COD2	Configuration-Control Specification(CCS)	DRAFT	監査対象となる DOC/HW/SW またはそれらへのポイントを定める
COD3	CAcert Official Documents Policy (COD)	破棄予定	ドキュメント形式を定める予定だったが、不要として破棄予定
COD4	Non-Related Persons – Disclaimer and Licence (NRP-DaL)	破棄済	外部の非メンバとの関係を規定する文書だったが、COD14(Root Distribution License) にてルート証明書の配布ライセンスを別途定めたので破棄された
COD5	Privacy Policy (PP)	POLICY	サイトおよび証明書の利用においての個人情報の扱いを規定する
COD6	Certification Practice Statement (CPS)	DRAFT	認証局および証明書の提供機能・管理方針・保証範囲など全側面に渡る運営方針を規定する
COD7	Dispute Resolution Policy (DRP)	POLICY	CAcert の運営およびメンバ間において発生する各種の係争に関する裁定方法を定める
COD8	Security Policy (SP)	DRAFT	システムおよび鍵の管理システムが耐障害性・安全性・可用性の各面で満たすべき事項を定める
COD9	CAcert Community Agreement (CCA)	有効	CAcert メンバーとして認証を受ける際に法的な束縛を行うための合意書
COD10	欠番	NA	
COD11	Oranisation Assurance Policy(OAP)	POLICY	組織体が CAcert 認証を受けるにあたって必要な手順と要件を定める
COD12	欠番	NA	
COD13	Assurance Policy (AP)	POLICY	認証の保証範囲、ポイントの付与基準、留意事項など認証活動の全範囲を規定する
COD14	Root Distribution License (RDL)	DRAFT	CAcert のルート証明書の配布ライセンスを規定する
-	Assurer Handbook	有効	認証手順および実施要件の実務解説

各文書は WiP(Work in Progress - 策定中)、DRAFT (草案段階)、POLICY (正式ポリシー) の三段階を経て策定されます。ちなみに、今年 (2010 年) はすべての文書が出揃い、CAcert の最終ゴールである証明局としての法的監査への準備が大きく進んだ年でした。

また、上記文書の中でも Member/Assurer の活動に特に密接なものを抜き出して関係を図示すると、以下のようになります：

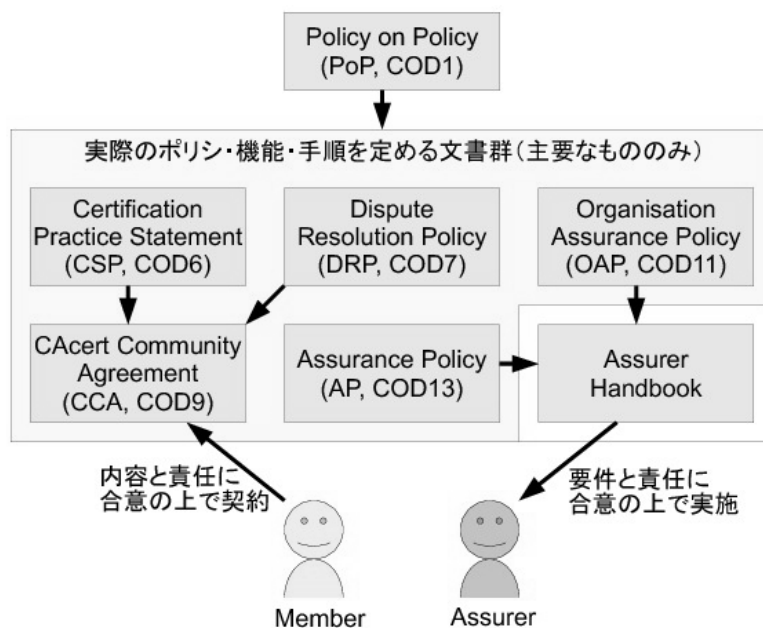


図 4 CAcert の主要ドキュメントの関係

各種文書では COD[0-9]+ や PoP/CCS/CCA/PP/DRL/... のような略称が頻繁に使われており初めて読むときは混乱してしまいがちです。まずは、図中の DRP(調停規則)/CCS(証明書運用規定)/CCA(会則)/Handbook(実務解説書)を押さえておくのがよいでしょう。

それでは次は組織構成です。問い合わせ先が見えにくいためまとめたのですが、実際に図にするとそう複雑でもなく、協同組合の CAcert Inc. を母体として、組織的な代表は通常組合員から選出・任命し、運営はコミュニティと共同という形態になっています。

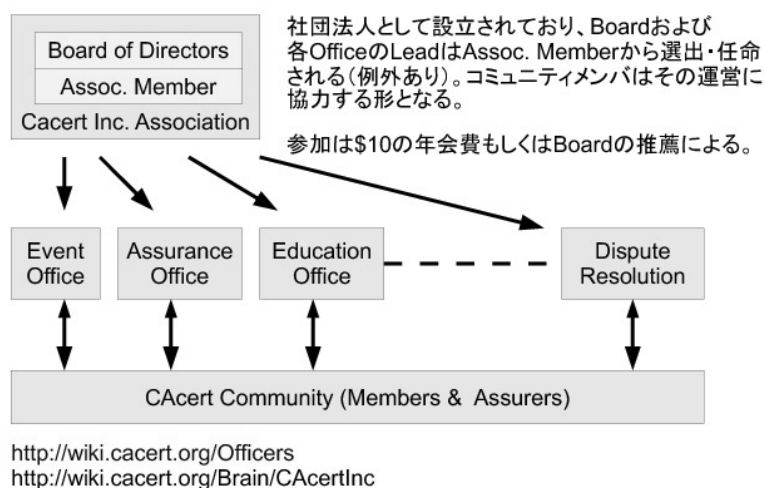


図 5 CAcert の組織構成 (REF: <http://wiki.cacert.org/Officers>)

なお、連絡を取る場合は CAcert Wiki で該当するオフィス(担当チーム)を探す、というのが「正しい」方法ですが、情報が散らばっていたりして判断できないこともあるので、

とにかく support@cacert.org に連絡して、振り分けしてもらおう

というのがよいでしょう。この ML はトリアーゼ役の人がウォッチしている（ことになっている）ので、適切な所に誘導して貰えるはずです。

6.7 まとめ

CAcert は組織も文書も、本格的な証明機関を目指しているだけあってかなり複雑になっていますが、利用するだけであればオンライン登録ですぐ証明書の発行を行えるので手軽に使うことができます。

これまでは欧州を中心^{*11}に展開してきているため日本ではそもそも Assurer が少ない状況ですが、CAcert も世界巡業をするなど徐々に展開を図っているようです。あなたも CAcert に参加してみませんか？

6.8 おまけ

この文書のために CAcert について調べなおしていたら、

1. Using Secure DNS to Associate Certificates with Domain Names For TLS
<https://datatracker.ietf.org/doc/draft-ietf-dane-protocol/>
2. 資料 - DNS に関わる技術動向 - KIDNS (Keys in DNS) -
<http://dnsops.jp/bof/20101125/201011-DNSOPS-KIDNSv5.pdf>

というものを芋蔓式に見つけました。

要はアクセス先から証明書が送られてきたら、アクセス先ドメインの CERT/TLSA レコードを引いて、そちらから得られた証明書（またはそのハッシュ）で検証する仕組みや、同様の手法を SSH/PGP に適用する仕組みになります^{*12}。これが普及したらドメインが絡むものについては、現行のルート CA からの信頼の階層という枠組み自体が不要になります。

実は 10 年近く前から検討されていたものの、最近ようやく DNSSEC が普及し始めた副産物として実用になってきたということですが、こういうものもあるということで、紹介しておきます。

^{*11} CAcert Inc. は豪州ですが、サーバの実体や活動は欧州が中心

^{*12} 実は GnuPG はすでに対応しているそうです … 知らなかった



7 俺の libsane が火を噴くぜ！

本庄 弘典

*13

SANE は Scanner Access Now Easy の頭文字を取ったもので、各種デバイスから画像を取り出すための汎用インターフェースです。主にスキャナから画像を取得するため、Linux や FreeBSD などフリーな OS で利用されます。今回は SANE の仕組みと libsane を用いたプログラムの作成方法について解説します。

7.1 SANE の構造

SANE を利用するアプリケーションは libsane.so という共有ライブラリをリンクします。libsane.so は dll と呼ばれる仮想デバイスドライバを通し、各デバイスごとに作られたドライバにアクセスすることでデバイスを制御します。また図に示されるように、net 仮想ドライバで他のマシンの saned に接続することで、リモートスキャナへのアクセスを実現します。[1, 2, 3]

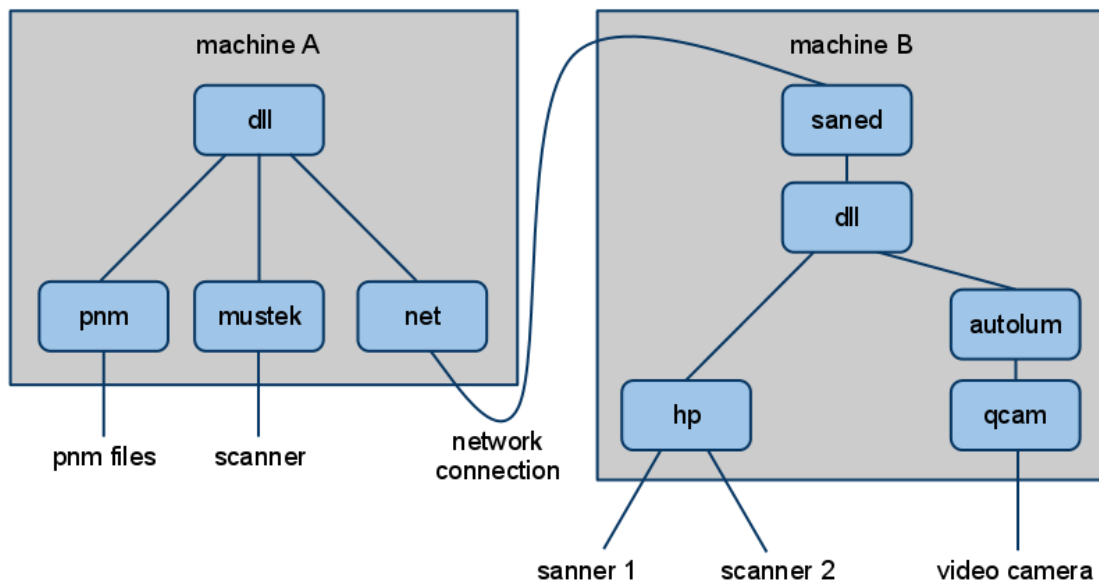


図 6 SANE の構造

なお、SANE を利用するユーザは scanner グループに所属させる必要があります。

*13 このタイトルは誰が付けたのでしょうか？

7.2 Code Flow

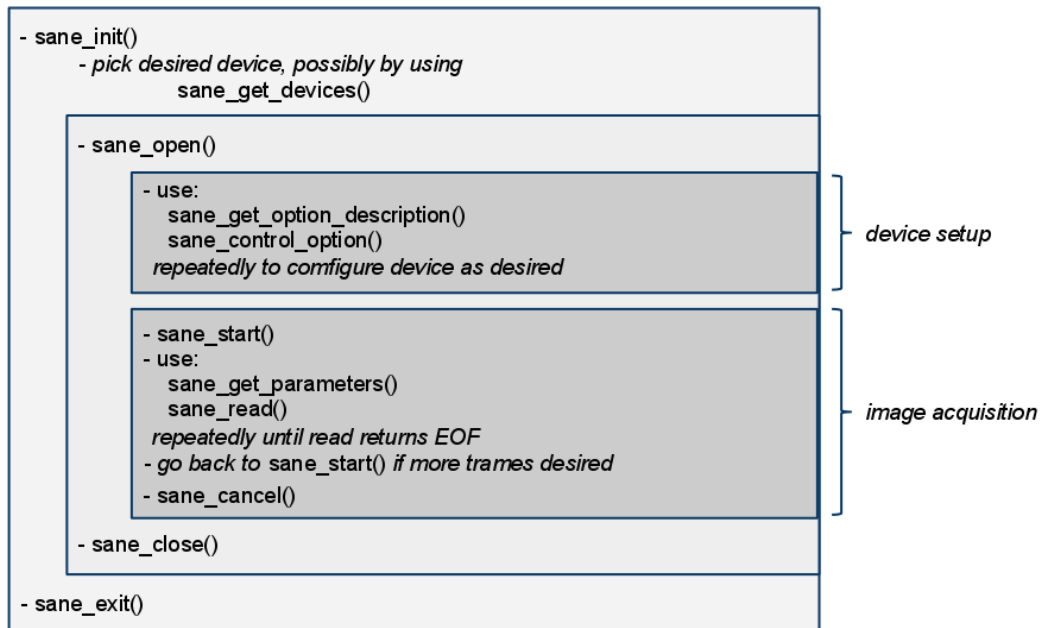


図 7 Code Flow

7.3 スキャンするためのコード

とりあえずスキャンするだけのコードは次のようになります。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include <sane/sane.h>

#define MAXLEN 32768

int main()
{
    SANE_Status stat;
    SANE_Handle hndl;
    SANE_Byte buf[MAXLEN];
    SANE_Int len = MAXLEN;
    FILE *fp;

    stat = sane_init(NULL, NULL);
    printf("init: %d\n", stat);
    stat = sane_open("fujitsu:libusb:008:002", &hndl);
    printf("open: %d\n", stat);

    fp = fopen("foo.bin", "wb");
    stat = sane_start(hndl);
    printf("start: %d\n", stat);
    while (len == MAXLEN) {
        stat = sane_read(hndl, buf, MAXLEN, &len);
        printf("read: %d\n", stat);
        printf("read_len: %d\n", len);
        fwrite(buf, len, 1, fp);
    }
    fclose(fp);

    sane_close(hndl);
    printf("close: %d\n", stat);
    sane_exit();
    printf("exit: %d\n", stat);
}
```

このコードで保存される foo.bin は、ScanSnap S1500 の場合、ヘッダ無しの pbm ファイルとして保存されました。スキャナによっては pgm で保存されるものもあり、デフォルトで保存される画像のモードは特に決まっていないようです。

7.4 カラーのスキャン

オプションの設定にはいくつか注意点があるようです。

- `sane_get_option_descriptor()` でオプションの説明文やフォーマットの取得が可能。
- `sane_control_option(hndl, 0, SANE_ACTION_GET_VALUE, &num, &info)` でオプションの総数を取得できる。
- `sane_control_option()` で Action に `SANE_ACTION_GET_VALUE` を指定することで現在設定されているオプションを読み出すことが可能。
- `sane_control_option()` で Action に `SANE_ACTION_SET_VALUE` を指定することでオプションの設定が可能。
- `sane_control_option()` を呼ぶ際には事前に `sane_get_option_descriptor()` を呼び出しておく必要がある。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sane/sane.h>

#define MAXLEN 32768

int main()
{
    SANE_Status stat;
    SANE_Handle hndl;
    const SANE_Option_Descriptor *opt;
    SANE_Int info = 0;
    SANE_Byte buf[MAXLEN];
    SANE_Int len = MAXLEN;
    SANE_Parameters param;
    FILE *fp;

    stat = sane_init(NULL, NULL);
    printf("stat: %d\n", stat);
    stat = sane_open("fujitsu:libusb:008:004", &hndl);
    printf("open: %d\n", stat);

    int x=300, y=300;
    opt = sane_get_option_descriptor(hndl, 4);
    printf("get_option_descriptor: %03d: %d\n", 0, opt->size);
    stat = sane_control_option(hndl, 4, SANE_ACTION_SET_VALUE, &x, &info);
    printf("control_option: %d\n", stat);
    printf("info: %d\n", info);

    opt = sane_get_option_descriptor(hndl, 5);
    printf("get_option_descriptor: %03d: %d\n", 0, opt->size);
    stat = sane_control_option(hndl, 5, SANE_ACTION_SET_VALUE, &y, &info);
    printf("control_option: %d\n", stat);
    printf("info: %d\n", info);

    strcpy(buf, "Color");
    opt = sane_get_option_descriptor(hndl, 3);
    printf("get_option_descriptor: %03d: %d\n", 0, opt->size);
    stat = sane_control_option(hndl, 3, SANE_ACTION_SET_VALUE, buf, &info);
    printf("control_option: %d\n", stat);
    printf("info: %d\n", info);

    stat = sane_start(hndl);
    printf("%d\n", stat);
    fp = fopen("foo.ppm", "wb");

    stat = sane_get_parameters(hndl, &param);
    printf("get_parameters: %d\n", stat);
    sprintf(buf, "P6\n# SANE data follows\n%d %d\n",
            param.pixels_per_line, param.lines, 255);
    fwrite(buf, strlen(buf), 1, fp);

    while (len == MAXLEN) {
        stat = sane_read(hndl, buf, MAXLEN, &len);
        printf("read: %d\n", stat);
        printf("read_len: %d\n", len);
        fwrite(buf, len, 1, fp);
    }
    fclose(fp);

    sane_close(hndl);
    printf("close: %d\n", stat);
    sane_exit();
    printf("exit: %d\n", stat);
}
```



図 8 スキャン結果

モノクロではわかりづらいと思いますが、ScanSnap S1500 でスキャンした結果、R と B が入れ替わってスキャンされました。

7.5 まとめ

- やや不安定 (特に権限周り)
 - scanimage -L でたまに segmentation fault する
- スキャナによってデフォルトの動作がちまちま
 - ScanSnap は.pbm でスキャン
 - CanoScan は.pgm でスキャン
- スキャナによって RGB の順番すらまちまち
 - でもこれは ScanSnap が悪いのかも
 - ScanSnap は TWAIN 未対応だし
- ドキュメント化されていない罫の存在
- スキャナのボタンをイベントとして使えないっぽい

参考文献

- [1] SANE - Scanner Access Now Easy
<http://www.sane-project.org/>
- [2] SANE-tutorial-JP / 著者:David Mosberger / 翻訳:川岸 良治
<http://archive.linux.or.jp/JF/JFdocs/SANE-tutorial-JP.html>
- [3] SANE Standard Version 2.0 proposal 0.08 - rauch/beck
<http://www.sane-project.org/sane2/0.08/>



Debian 勉強会資料

2010年12月18日 初版第1刷発行

東京エリア Debian 勉強会（編集・印刷・発行）
