FreeBSD 勉強会

# 使ってみよう分散ファイルシステム（1）
# OpenAFSの巻

佐藤　広生 <hrs@FreeBSD.org>

東京工業大学/ FreeBSD Project

2014/11/13

# 講師紹介

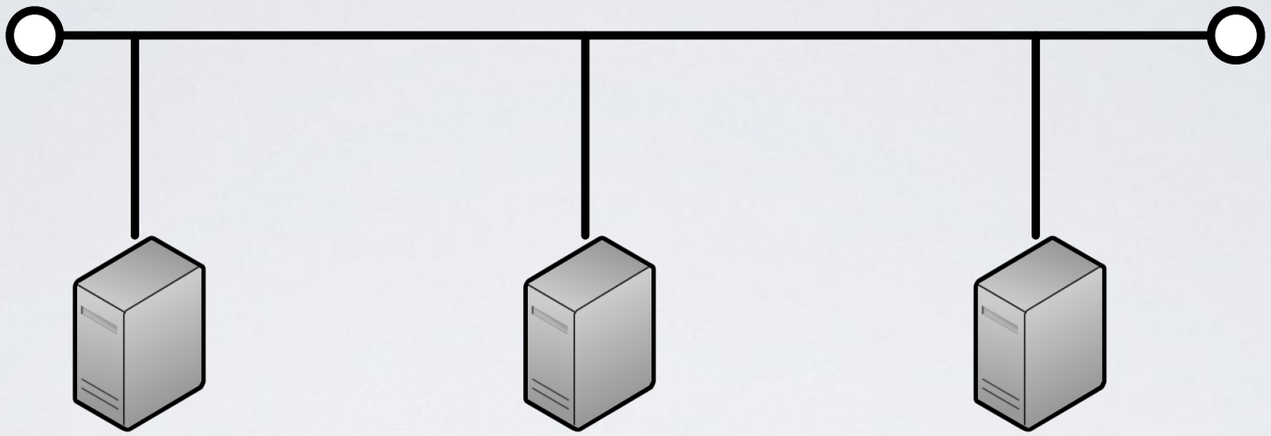
- 佐藤広生（さとうひろき）
  - FreeBSD src/ports/doc committer
  - FreeBSDコアチームメンバ(2006より5期目), リリースエンジニア
  - FreeBSD Foundation理事(2008-)
  - 東京工業大学 助教

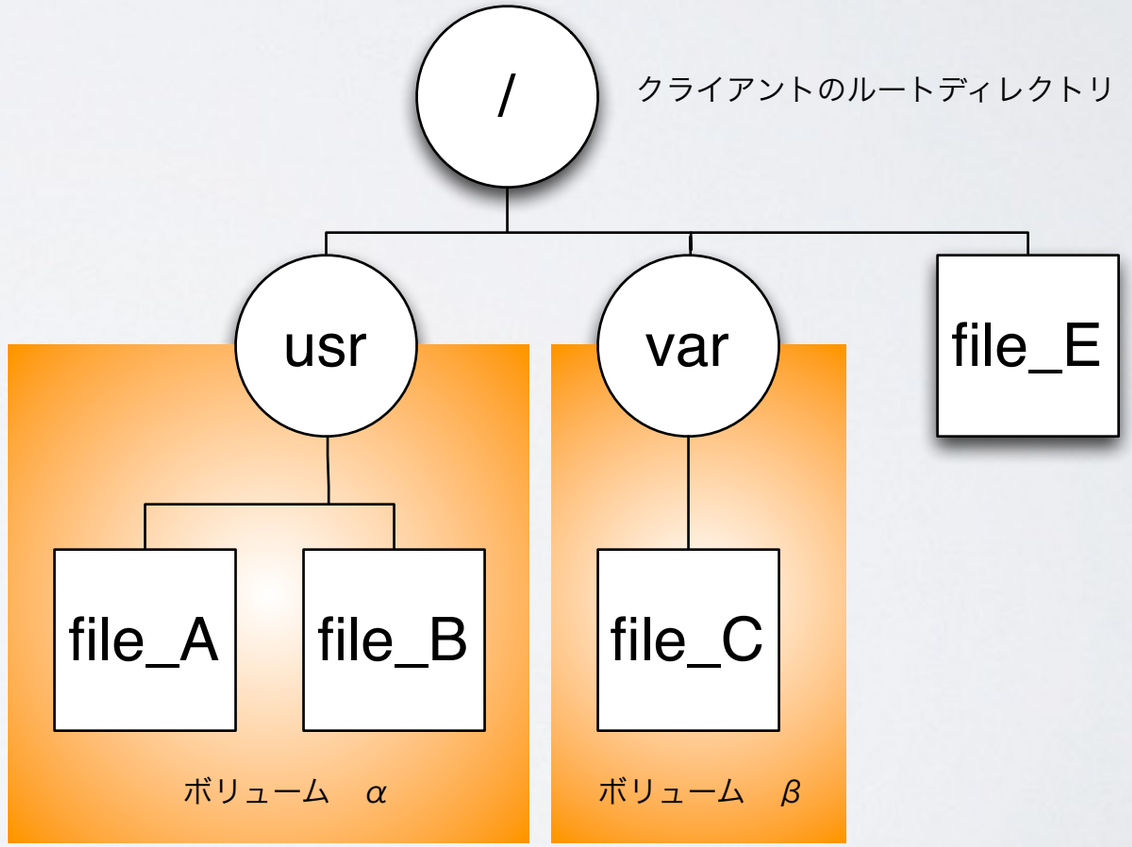
- 技術的なご相談は hrs@allbsd.org まで

# お話すること

- **分散ファイルシステム**
  - どんなもの？
- **OpenAFS**
  - 背景知識
  - 設定してみよう
    - Kerberos
    - DNS
    - シングルボリューム構成で動かすには
- 実用的な構成方法は、次回に解説します

# 分散ファイルシステム




複数のファイルサーバを使って
ひとつのファイルシステムを
見せるための仕組み

# 分散ファイルシステム

- 動機：多数のクライアントに同一の「ビュー」を提供したい。
- NFSじゃダメなのか？
  - サーバにデータがある
  - クラインアントがそのデータにアクセス
  - 多数あると？
    - サーバの負荷が上がる
    - サーバを増やす？ → サーバ間のデータ同期や一貫性は？

# 分散ファイルシステム

- 動機：多数のクライアントに同一の「ビュー」を提供したい。
- 設計目標：「スケーラビリティ」
  - 10,000台程度のクライアントにサービスできること
  - サーバの台数を増やすことができること
  - 多数のサーバ・クライアント管理が容易であること

# Andrew File System

- 1983年にカーネギーメロン大学で始まったプロジェクト
- 4.2BSD
- 1989年にTransarc社が商用化、1994年にIBMが買収
- DFSという名前でDCEに含まれるようになった（IBM AIX, Solaris, HP-UX等）
- 2000年にIBMがOpenAFSとしてオープンソース化
- Arla, Coda 等の派生プロジェクトが存在するが....

# Andrew File System

- 米国大学の学生用ホームディレクトリの提供などに
  現在も使われている。

Stanford | University IT
*Administrative Systems*

Google™ Custom Search

FIND SERVICES ▾ I WANT TO ... ▾ GET HELP ▾ SECURITY ▾ ABOUT US

## File and Data Storage: AFS

AFS (Andrew File System) is a distributed, networked file system that enables efficient file sharing between clients and servers. AFS files are accessible via the Web or through file transfer programs such as OpenAFS or Fetch (Macintosh) and SecureFX (Windows).

**Currently all users with a full-service SUNet ID are granted 5 GB of AFS file space. Additional disk space is available by request for faculty-sponsored research including dissertations.**

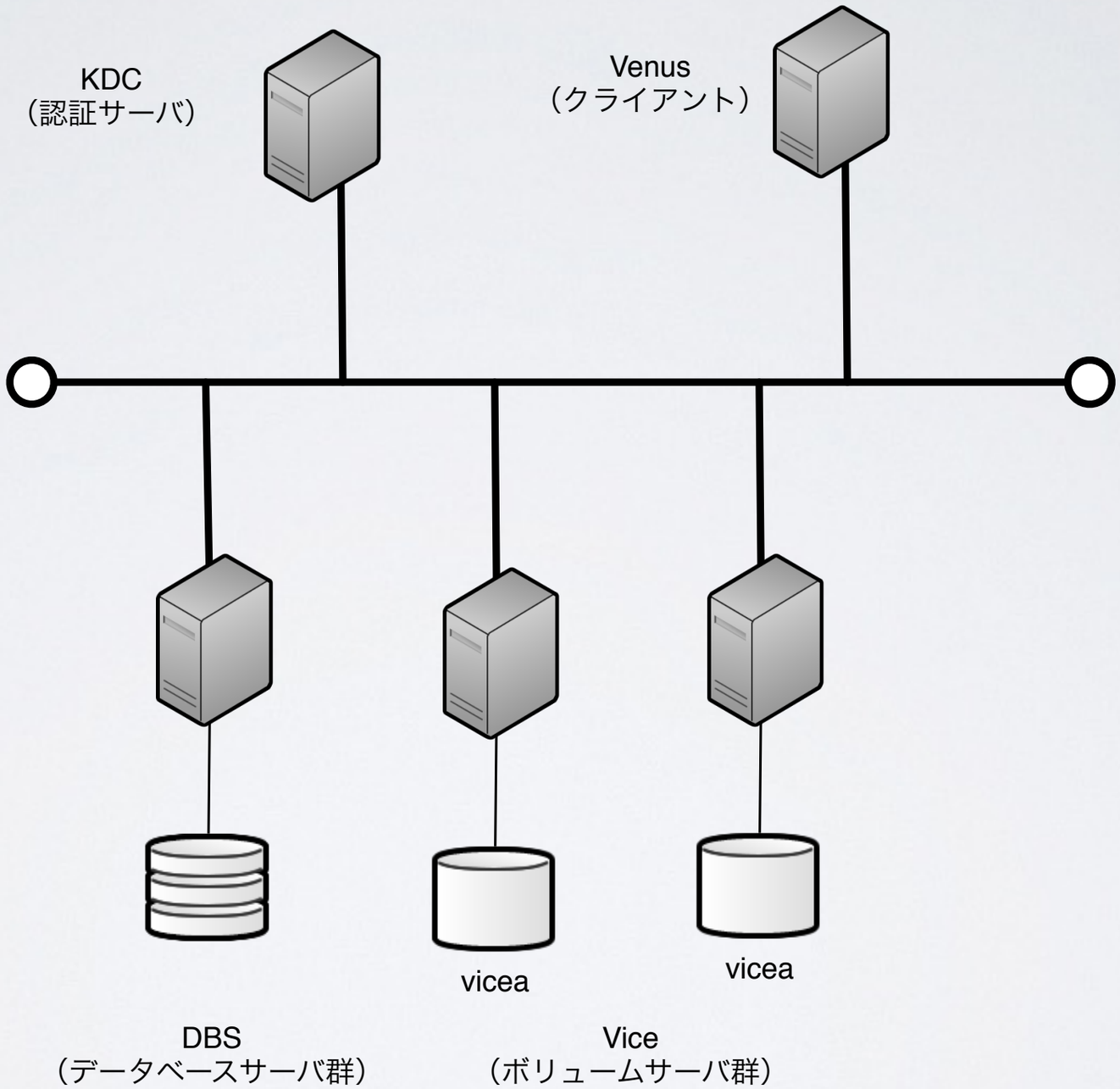
Request AFS Services

Launch AFS on the Web (WebAFS)

AT A GLANCE

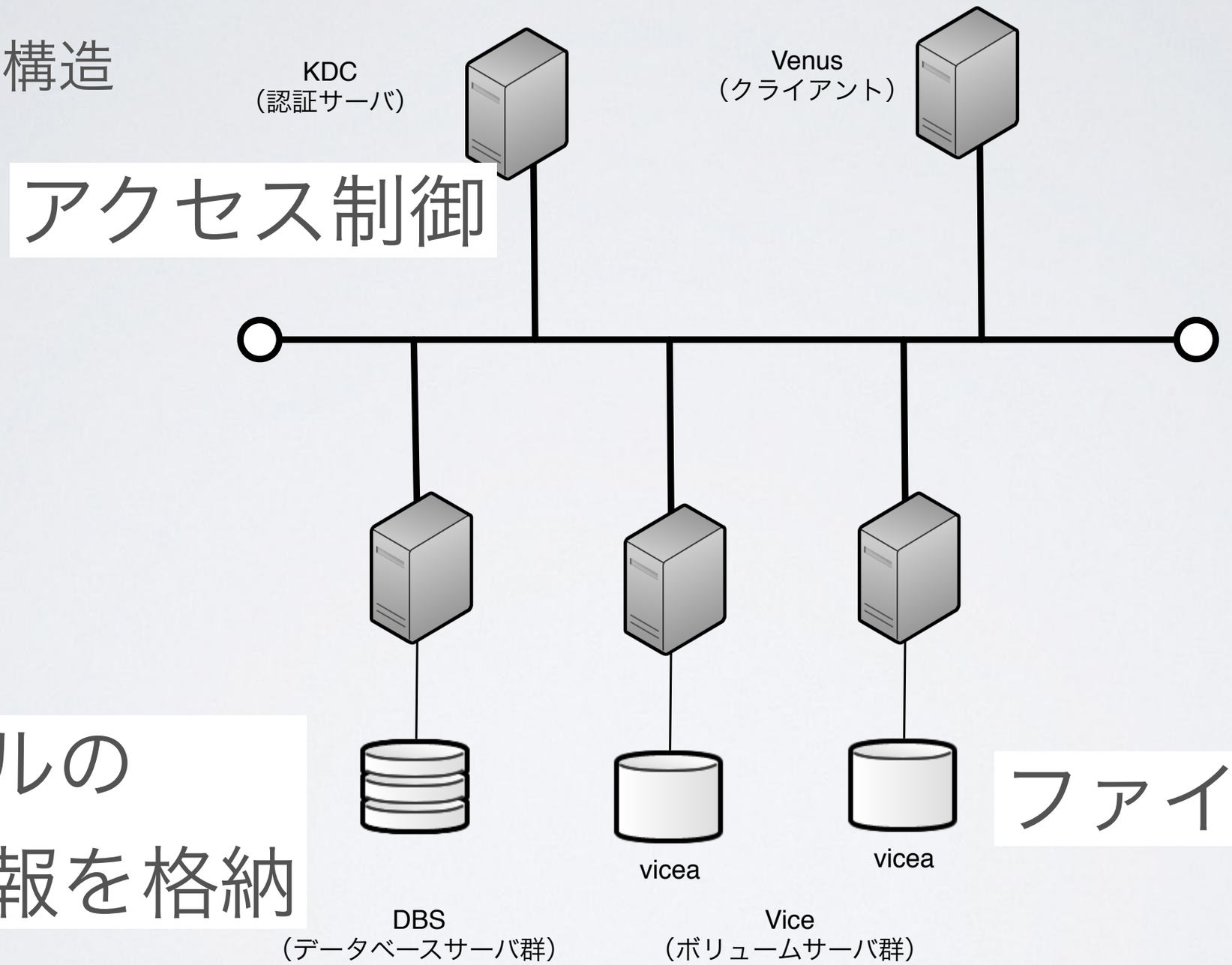
# Andrew File System

- 基本構造



# Andrew File System

- 基本構造



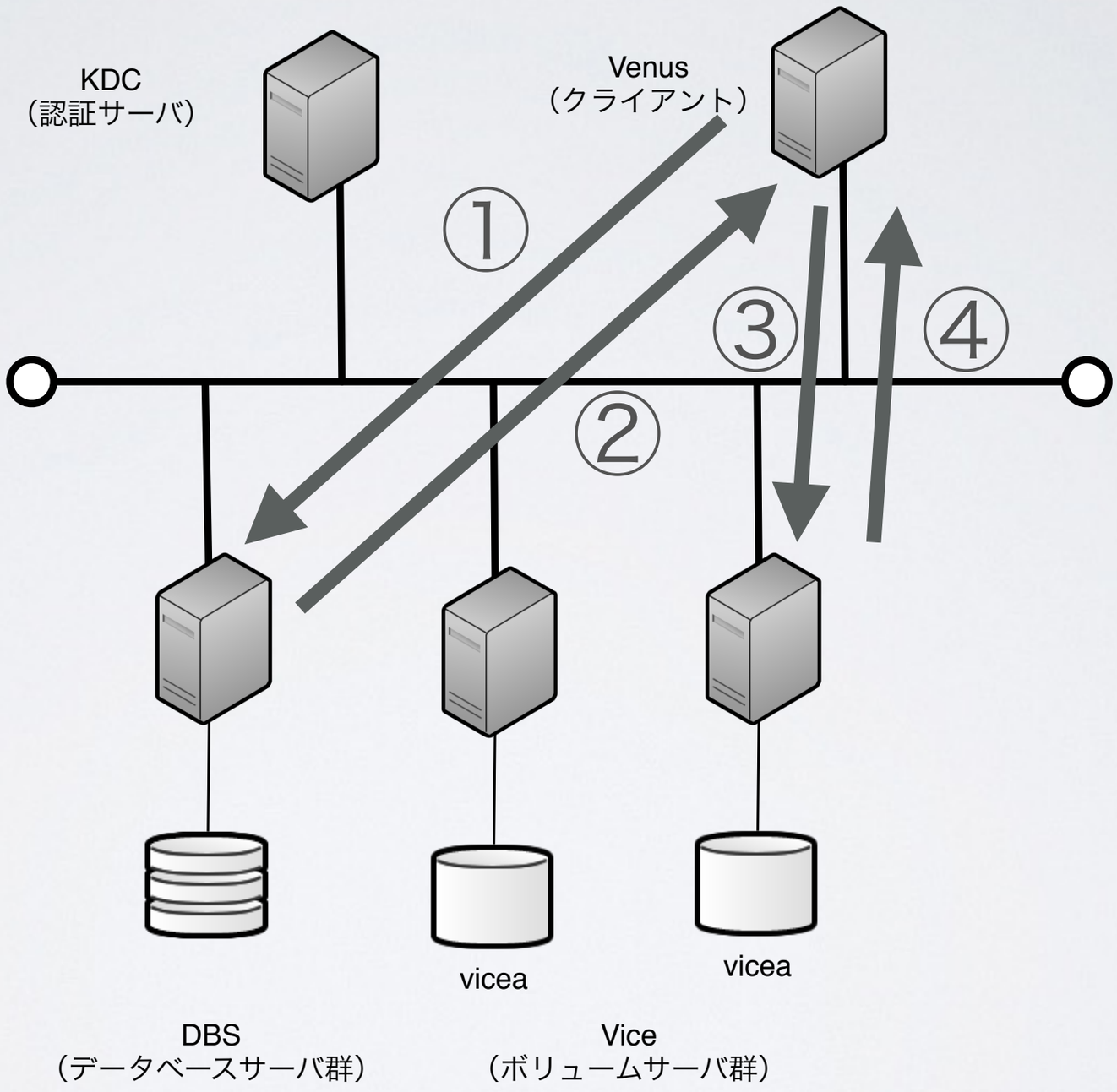

アクセス制御

ファイルの
位置情報を格納

ファイルを格納

# Andrew File System

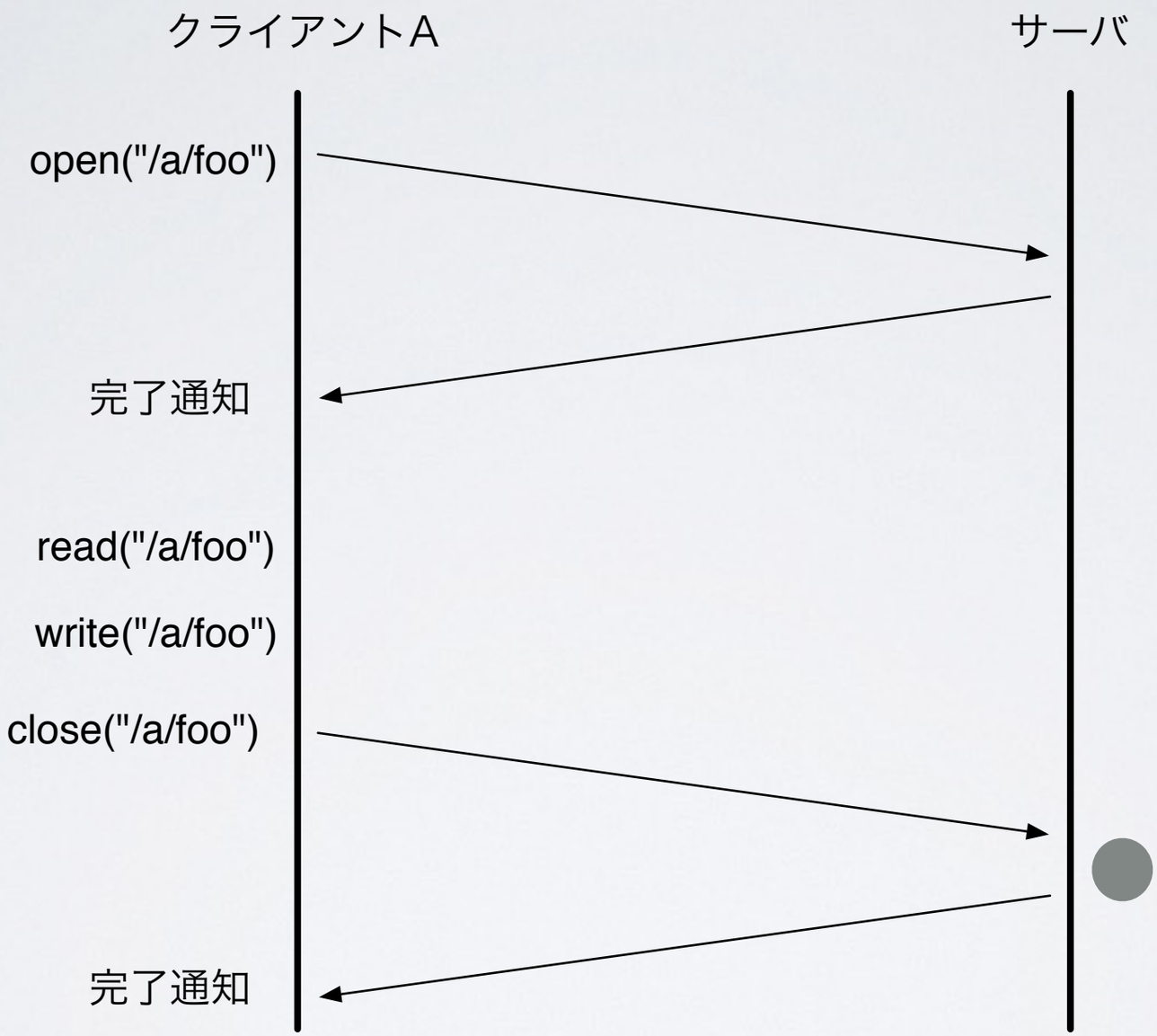
- 基本構造



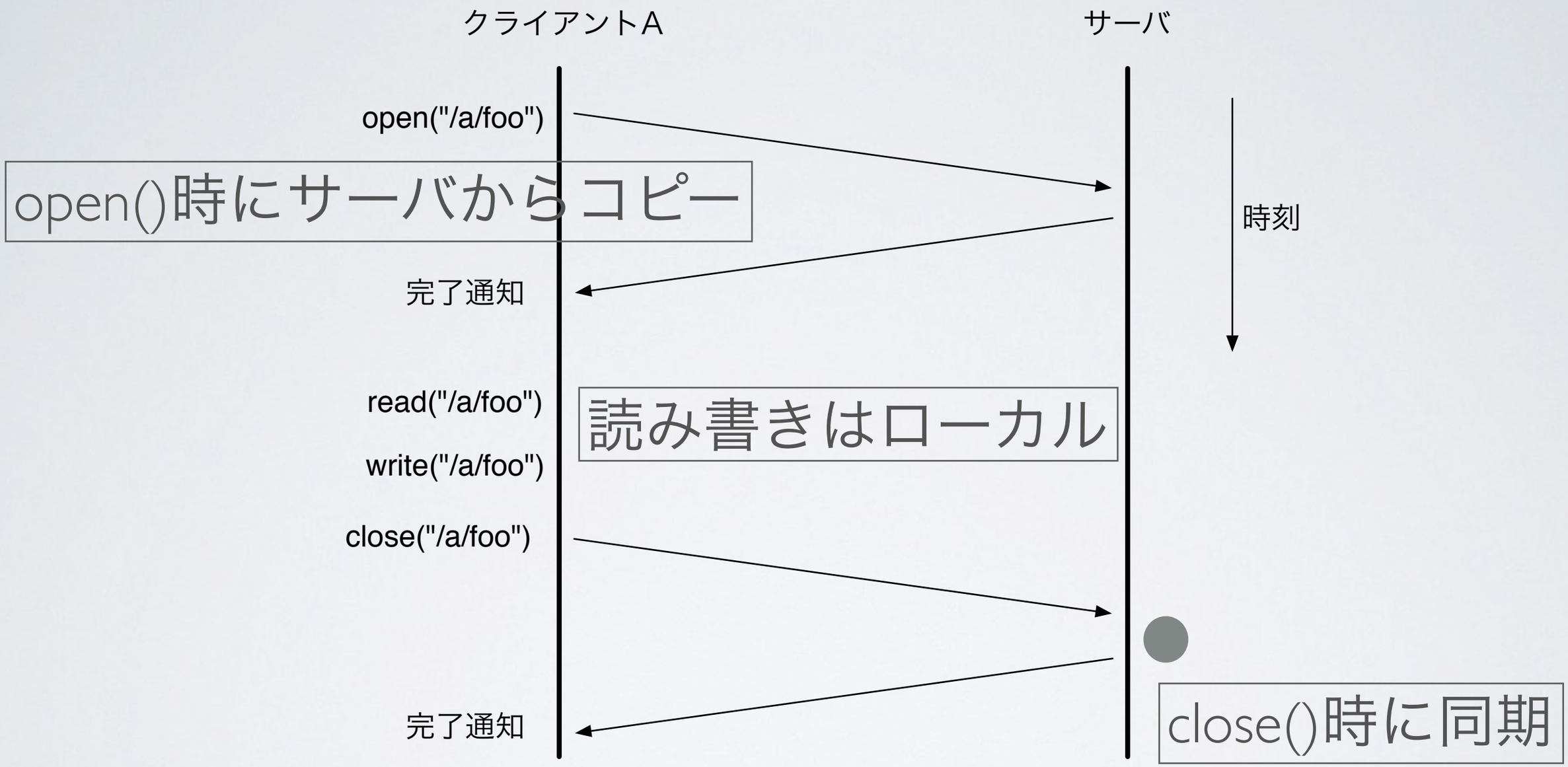
# NFSとの比較

| | AFS | NFS |
|---|---|---|
| ファイル名前空間 | 単一 | サーバに依存 |
| ファイルの位置 | 自動検出 | マウントポイント |
| キャッシュ | クライアント、一貫性保証 | クライアント |
| セキュリティ | KerberosベースのACL | UID(v2,v3), ACL(v4) |
| 可用性 | ファイル、DB | なし |
| バックアップ | オンラインで可能 | なし |
| 構成変更 | オンラインで可能 | マウントポイント変更 |
| 管理 | どこからでも可能 | サーバからのみ |

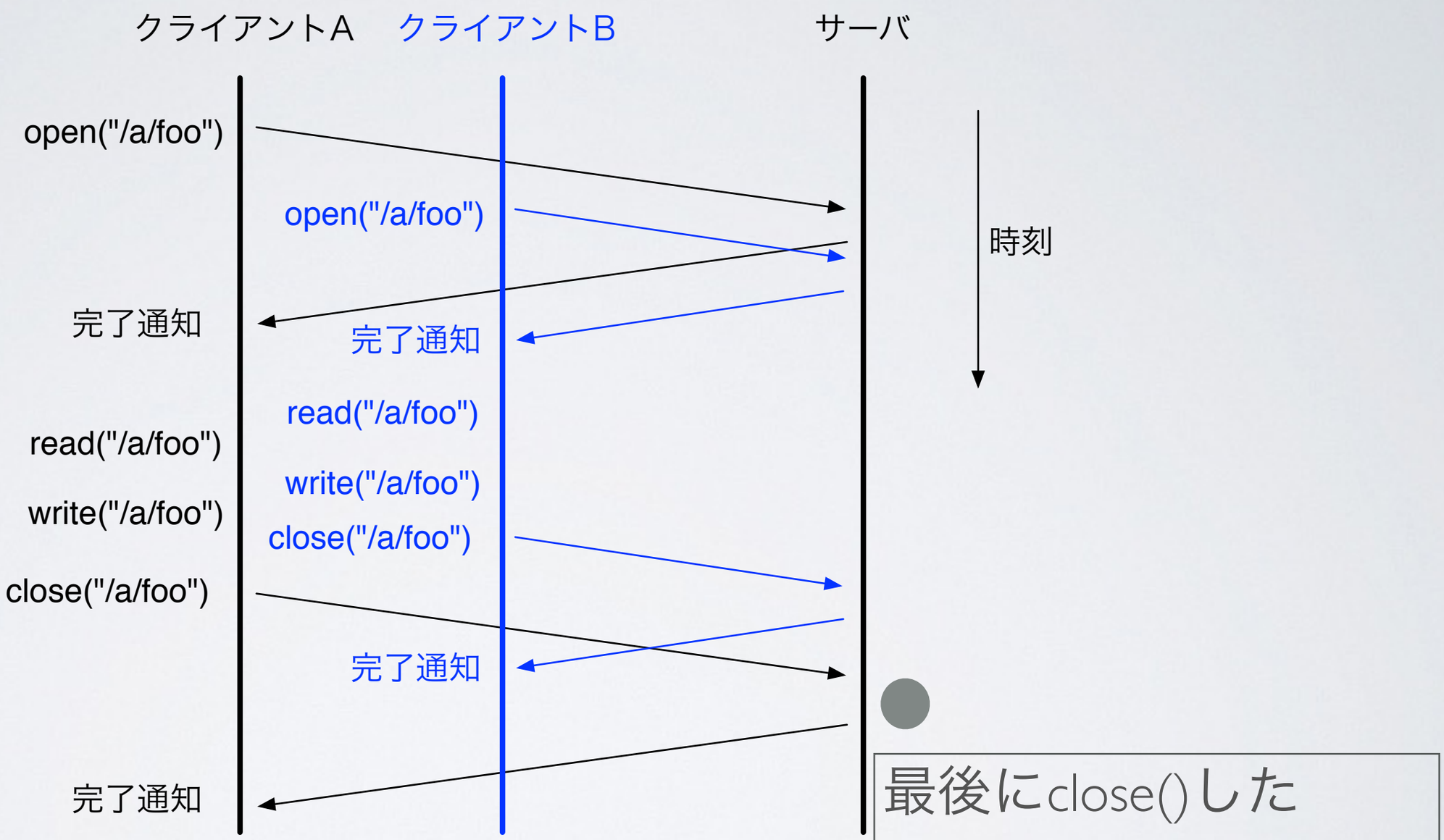
# AFSのセマンティクス



# AFSのセマンティクス



# AFSのセマンティクス




最後にclose()したクライアントのデータで上書きされる

# AFSのセマンティクス

- open()する時に、サーバに登録する（callback）
- サーバは、データ変更時にcallback登録した
  クライアントに通知（callback登録の破棄）

- クライアントはopen()する時、次のように振る舞う
  - キャッシュにある＋callback登録あり＝キャッシュを使う
  - キャッシュにある＋callback登録なし＝取り直す
  - キャッシュになし＋callback登録なし＝取り直す

- キャッシュを有効利用し、スケーラビリティを向上

# AFSのセマンティクス

- NFSには、こういった一貫した挙動がない
  - クライアントキャッシュ（通常３０秒ほど）
    - サーバで更新されたデータがいつ見えるか不明
      - 一定時間が過ぎると、毎回サーバに問い合わせる
  - 複数のクライアントからの書き込みが混ざる
- 動くけれどもスケールしない

# 性能の理論計算

| シーケンシャル読出 | AFS | NFS |
|---|---|---|
| 小ファイル（初回） | ブロック数×接続遅延時間 | |
| 小ファイル | ブロック数×キャッシュ遅延時間 | |
| 大ファイル（初回） | ブロック数×接続遅延時間 | |
| 大ファイル | ブロック数×キャッシュ遅延 | ブロック数×接続遅延時間 |

# 性能の理論計算

| シーケンシャル読出 | AFS | NFS |
|---|---|---|
| 小ファイル（初回） | ブロック数×接続遅延時間 | |
| 小ファイ | | |
| 大ファイ | | |
| 大ファイル | ブロック数×キャッシュ遅延 | ブロック数×接続遅延時間 |

AFSは、callbackが破棄されていなければ
大きいファイルでも再open()時に
キャッシュが効く

# 性能の理論計算

| シーケンシャル書込 | AFS | NFS |
|---|---|---|
| 小ファイル | ブロック数×接続遅延時間 | |
| 大ファイル | ブロック数×接続遅延時間 | |
| 大ファイル上書き | 2×ブロック数×接続遅延時間 | ブロック数×接続遅延時間 |
| 大ファイル一部書込 | 2×ブロック数×接続遅延時間 | 1×接続遅延時間 |

# 性能の理論計算

| シーケンシャル書込 | AFS | NFS |
|---|---|---|
| 小ファイル | | |
| 大ファイル | | |
| 大ファイル上書き | 2×ブロック数×接続遅延時間 | ブロック数×接続遅延時間 |
| 大ファイル一部書込 | 2×ブロック数×接続遅延時間 | 1×接続遅延時間 |

クライアントは一度必ずファイル全体を読まないといけない

# つかってみよう

- **FreeBSD 上での OpenAFS の設定**
  - Heimdal（Kerberos）
  - BIND（DNS）
  - OpenAFS サーバ
  - OpenAFS クライアント

# Heimdal

- **認証のために必ず必要**
- http://people.allbsd.org/~hrs/FreeBSD/sato-20140313.pdf

```
# kadmin -l init ALLBSD.ORG
Realm max ticket life [unlimited]:
Realm max renewable ticket life [unlimited]:
# kadmin -l stash
Master key:
Verifying - Master key:
kadmin: writing key to "/var/heimdal/m-key"
```

①レルムを決めて初期化

```
# kadmin -l add hrs/admin@ALLBSD.ORG
Max ticket life [1 day]:
Max renewable life [1 week]:
Principal expiration time [never]:
Password expiration time [never]:
Attributes []:
hrs/admin@ALLBSD.ORG's Password:
Verifying - hrs/admin@ALLBSD.ORG's Password:
```

②管理用プリンシパル追加

# Heimdal

- **認証のために必ず必要**
- http://people.allbsd.org/~hrs/FreeBSD/sato-20140313.pdf

```
kerberos5_server_enable="YES"
kadmind5_server_enable="YES"
kpasswdd_server_enable="YES"
```

③rc.confを編集

```
# service kerberos start
# service kadmind start
# service kpasswdd start
```

④デーモンの起動

⑤/var/heimdal/kadmind.aclの編集

```
hrs/admin@ALLBSD.ORG all *
```

# Heimdal

- **認証のために必ず必要**
- http://people.allbsd.org/~hrs/FreeBSD/sato-20140313.pdf

⑥DNSに次のエントリを追加

```
_kerberos._tcp                  SRV     10 1 88 kdc.allbsd.org.
_kerberos._udp                  SRV     10 1 88 kdc.allbsd.org.
_kerberos._tcp                  SRV     20 1 88 kdc2.allbsd.org.
_kerberos._udp                  SRV     20 1 88 kdc2.allbsd.org.

_kpasswd._udp                   SRV     10 1 464 kdc.allbsd.org.
_kerberos-adm._tcp              SRV     10 1 749 kdc.allbsd.org.
_kerberos                       TXT     "ALLBSD.ORG"
```

# OpenAFSサーバ

## OpenAFSのインストール

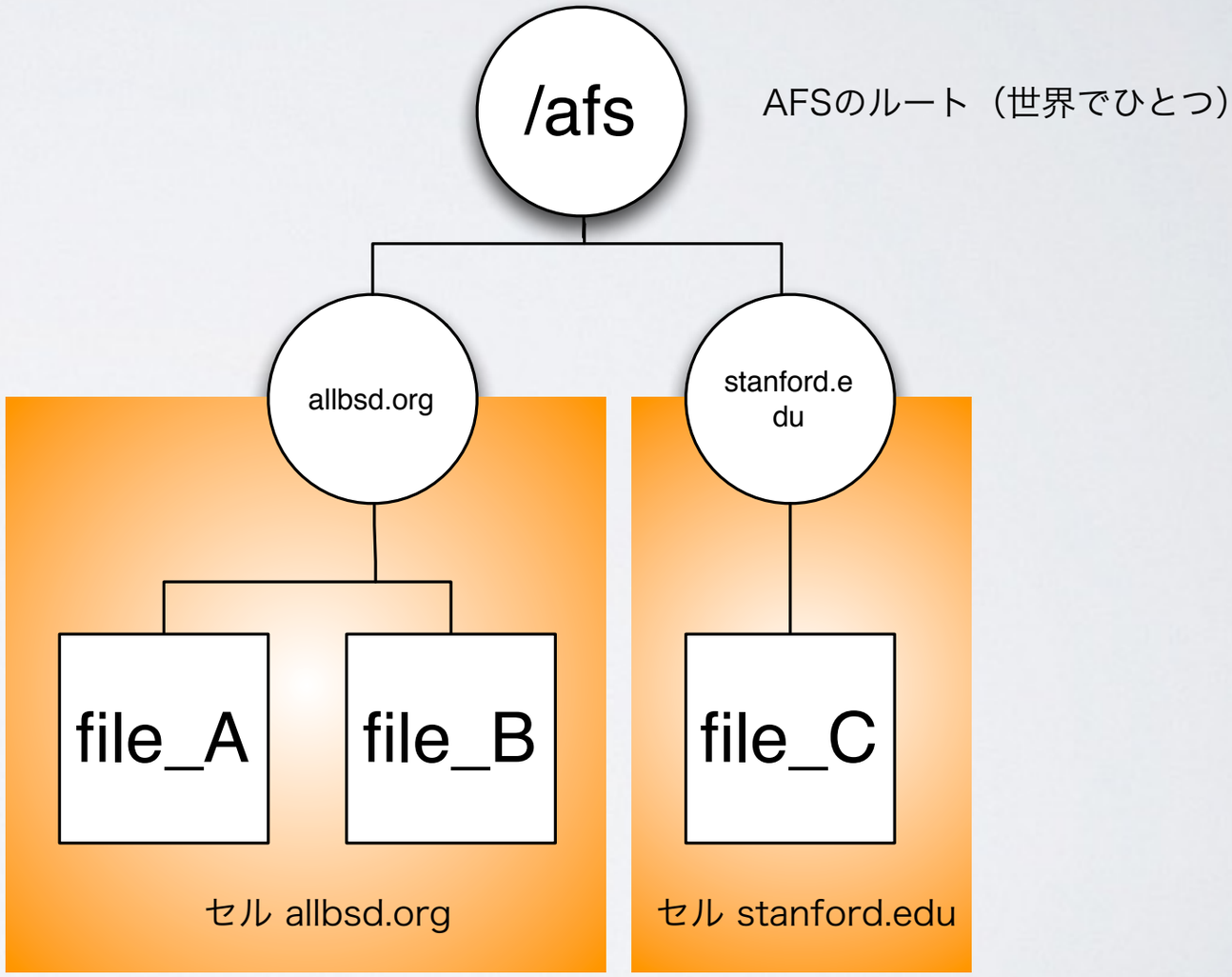
```
# cd /usr/ports/net/openafs
# make install
```

/usr/local/etc/openafs: 設定ファイル群

/var/openafs: データベースファイル群

# OpenAFSサーバ

- まずはセル名を決めよう
  - AFSは単一の名前空間を持っている（世界で唯一）
  - /afs/allbsd.org/....



# OpenAFSサーバ

AFSセルの名前を設定

サンプルはopenafs.org になっているので必ず変える。

デフォルトの /usr/local/etc/openafs にあるファイルを、

/usr/local/etc/openafs/server に移動させる。

/usr/afs/etc に /usr/local/etc/openafs/server を指すsymlinkを置く

```
# mkdir /usr/local/etc/openafs/server
# mv /usr/local/etc/openafs/* /usr/local/etc/openafs/server
mv: rename server to server/server: Invalid argument
# mv /usr/local/etc/openafs/server/cacheinfo /usr/local/etc/openafs
# mkdir /usr/afs
# ln -s /usr/local/etc/openafs/server /usr/afs/etc
# chmod 0770 /var/openafs

# echo "allbsd.org" > /usr/local/etc/openafs/ThisCell
```

# OpenAFSサーバ

セルを構成するAFSサーバのRRをDNSに登録する
（ここでは fs.allbsd.org というマシン）

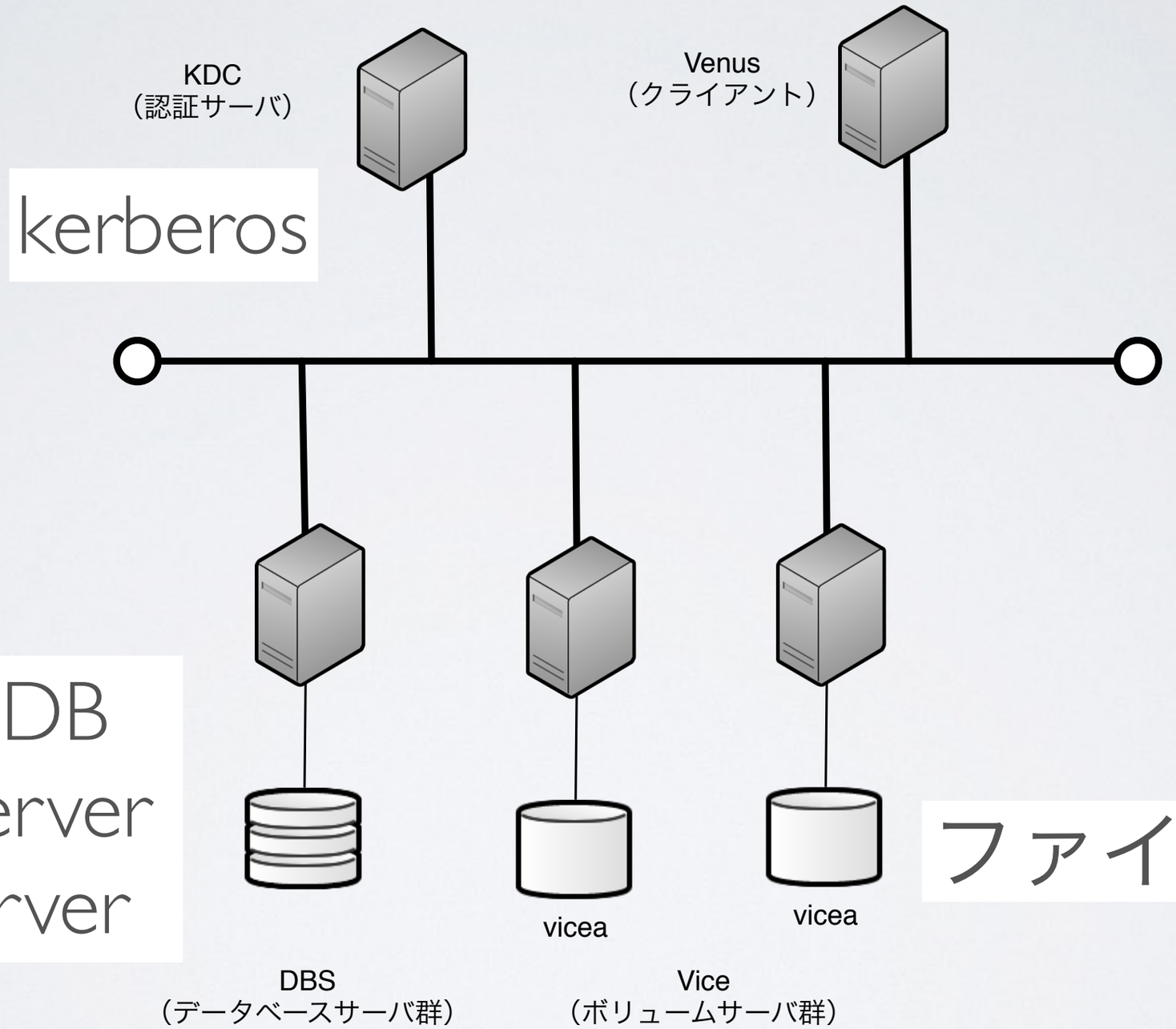
```
@                               IN AFSDB 1              fs.allbsd.org.

_afs3-vlserver._udp             SRV     0 1 7003        fs.allbsd.org.
_afs3-vlserver._tcp             SRV     0 1 7003        fs.allbsd.org.

_afs3-prserver._udp             SRV     0 1 7002        fs.allbsd.org.
_afs3-prserver._tcp             SRV     0 1 7002        fs.allbsd.org.
```

prserver: ユーザデータベース(protection DB)

vlserver: ボリューム位置データベース (volume location DB)

# OpenAFSサーバ

- 基本構造




AFSDB
prserver
vlserver

ファイルを格納

# OpenAFSサーバ

AFSセル用のサービスプリンシパルを追加

```
% kadmin -p hrs/admin add --random-key --use-defaults afs/allbsd.org@ALLBSD.ORG
hrs/admin@ALLBSD.ORG's Password:
```

# OpenAFSサーバ

## 作成した鍵をKDCから取り出す

AFSKEYFILE: という書式で kadmin ext を実行すると取り出せる。

この形式でないとOpenAFSは認識しないので注意！

```
# kadmin -p hrs/admin ¥
  ext -k AFSKEYFILE:/usr/local/etc/openafs/server/KeyFile afs/allbsd.org@ALLBSD.ORG
hrs/admin@ALLBSD.ORG's Password:
```

# OpenAFSサーバ

作成した鍵をKDCから取り出す

ちゃんと作成できているかどうか、asetkey list というコマンドで確認できる。

```
# asetkey list
kvno    1: key is: 918c5858b562732c
All done.
```

# OpenAFSサーバ

## BOSサーバを起動する

「Basic Overseer Server」。セルの管理を担当するサーバ

まずセル名を設定、DBサーバが認識されているかどうか確認。

（DNS設定が間違っていると、ここでHostが出てこない）

```
# /usr/local/sbin/bosserver

# bos setcellname -server fs.allbsd.org -name allbsd.org -localauth

# bos listhosts fs2.allbsd.org
bos: running unauthenticated
Cell name is allbsd.org
    Host 1 is [fs2.allbsd.org]
```

# OpenAFSサーバ

BOSの管理権限を持つユーザを追加する

管理用に"sysadmin" を追加すると良い。

管理権限を持つ一般ユーザは、

区別のために"ユーザ名.afs"とする。

```
# bos adduser fs.allbsd.org sysadmin -localauth
# bos adduser fs.allbsd.org hrs.afs -localauth
# bos listusers fs2.allbsd.org
bos: running unauthenticated
SUsers are: sysadmin hrs.afs
```

# OpenAFSサーバ

サーバインスタンスを追加する

BUサーバ、PTサーバ、VLサーバを起動するように設定

```
# bos create fs.allbsd.org buserver simple ¥
    /usr/local/libexec/openafs/buserver -localauth
# bos create fs.allbsd.org ptserver simple ¥
    /usr/local/libexec/openafs/ptserver -localauth
# bos create fs.allbsd.org vlserver simple ¥
    /usr/local/libexec/openafs/vlserver -localauth
# bos status fs.allbsd.org
bos: running unauthenticated
SUsers are: sysadmin hrs.afs
```

# OpenAFSサーバ

サーバインスタンスを追加する

bos status コマンドで状況を確認できる

```
# bos status fs.allbsd.org -long -localauth
Instance buserver, (type is simple) currently running normally.
    Process last started at Thu Nov 13 15:50:55 2014 (1 proc starts)
    Command 1 is '/usr/local/libexec/openafs/buserver'

Instance ptserver, (type is simple) currently running normally.
    Process last started at Thu Nov 13 15:51:40 2014 (1 proc starts)
    Command 1 is '/usr/local/libexec/openafs/ptserver'

Instance vlserver, (type is simple) currently running normally.
    Process last started at Thu Nov 13 15:51:47 2014 (1 proc starts)
    Command 1 is '/usr/local/libexec/openafs/vlserver'
```

# OpenAFSサーバ

BOSサーバを再起動する

```
afsserver_enable="YES"
```

```
# service afsserver restart
```

# OpenAFSサーバ

ここでチェック

/usr/local/etc/openafs/CellServDBのIPアドレスに[]が付いていたら外す

```
% cat /usr/local/etc/openafs/CellServDB
>allbsd.org     #Cell name
[192.168.2.1]     #fs.allbsd.org
```

# OpenAFSサーバ

PTデータベースに、AFS利用者を登録する

```
# pts createuser sysadmin -id 1 -localauth
# pts adduser sysadmin system:administrators -localauth

# pts createuser hrs.afs -id 21001 -localauth
# pts adduser hrs.afs system:administrators -localauth
```

# OpenAFSサーバ

PTデータベースに、AFS利用者を登録する

```
# pts listentries -user -localauth
Name                          ID  Owner Creator
anonymous                  32766   -204    -204
hrs.afs                    21001   -204    -204
sysadmin                       1   -204    -204

# pts listentries -group -localauth
Name                          ID  Owner Creator
system:administrators       -204   -204    -204
system:backup               -205   -204    -204
system:anyuser              -101   -204    -204
system:authuser             -102   -204    -204
system:ptsviewers           -203   -204    -204
```

# OpenAFSサーバ

PT

ここまでで、データベースサーバの設定は完了

すべての設定は/usr/local/etc/openafs/BosConfigに自動的に記録されている

# OpenAFSサーバ

AFSクライアントデーモンを起動

```
afsd_enable="YES"
```

```
# service afsd start
```

# OpenAFSサーバ

FSサーバインスタンスを追加する

3プロセスを登録する

```
# bos create fs.allbsd.org fs fs ¥
    -cmd /usr/local/libexec/openafs/fileserver ¥
    -cmd /usr/local/libexec/openafs/volserver ¥
    -cmd /usr/local/libexec/openafs/salvager -localauth
```

# OpenAFSサーバ

ボリュームをつくる

root.afs が /afs に、root.cell が /afs/allbsd.org に対応

```
# vos create fs.allbsd.org /vicepa root.afs -localauth
Volume 536870915 created on partition /vicepa of fs.allbsd.org
# vos create fs.allbsd.org /vicepa root.cell -localauth
Volume 536870915 created on partition /vicepa of fs.allbsd.org

# vos listvol fs.allbsd.org -localauth
Total number of volumes on server fs.allbsd.org partition /vicepa: 1
root.afs                          536870915 RW          2 K On-line
root.cell                          536870915 RW          2 K On-line

Total volumes onLine 1 ; Total volumes offLine 0 ; Total busy 0
```

# OpenAFSサーバ

DESのKerberos鍵を使えるようにしておくこと

(/etc/krb5.conf)

```
[libdefaults]
        allow_weak_crypto = TRUE
        default_etypes = des-cbc-crc
```

# OpenAFSサーバ

さてアクセス！

```
# cd /afs/allbsd.org
# ls -al
total 0
ls: .: Permission denied
```

# OpenAFSサーバ

さてアクセス！

```
% kinit hrs/afs@ALLBSD.ORG
hrs/afs@ALLBSD.ORG's Password:
% aklog -d
Authenticating to cell allbsd.org (server fs.allbsd.org).
Trying to authenticate to user's realm ALLBSD.ORG.
Getting tickets: afs/allbsd.org@ALLBSD.ORG
Using Kerberos V5 ticket natively
About to resolve name hrs.afs to id in cell allbsd.org.
Id 21001
Set username to AFS ID 21001
Setting tokens. AFS ID 21001 @ allbsd.org
# cd /afs/allbsd.org
# ls -al
total 5
drwxrwxrwx  2 root  wheel  2048 Nov 13 17:38 ./
drwxr-xr-x  5 root  wheel  2048 Jan  1  1970 ../
```

# OpenAFSサーバ

kinit で鍵を取得、aklog でAFS鍵に変換

あとは自由に読み書きできる

```
% kinit hrs/afs@ALLBSD.ORG
hrs/afs@ALLBSD.ORG's Password:
% aklog -d
Authenticating to cell allbsd.org (server fs.allbsd.org).
Trying to authenticate to user's realm ALLBSD.ORG.
Getting tickets: afs/allbsd.org@ALLBSD.ORG
Using Kerberos V5 ticket natively
About to resolve name hrs.afs to id in cell allbsd.org.
Id 21001
Set username to AFS ID 21001
Setting tokens. AFS ID 21001 @ allbsd.org
# cd /afs/allbsd.org
# ls -al
total 5
drwxrwxrwx  2 root  wheel  2048 Nov 13 17:38 ./
drwxr-xr-x  5 root  wheel  2048 Jan  1  1970 ../
```

# OpenAFSサーバ

アクセス制御

```
% fs listacl /afs/allbsd.org
Access list for /afs/allbsd.org is
Normal rights:
  system:administrators rlidwka

% fs setacl /afs/allbsd.org system:anyuser rl
% fs listacl /afs/allbsd.org
Access list for /afs/allbsd.org is
Normal rights:
  system:administrators rlidwka
  system:anyuser rl
```

# OpenAFSサーバ

2台目を追加するには？

- FSサーバだけを増やせばよい！
- 新しいサーバ(fs2.allbsd.org)：
  - bos setcellname して、FSインスタンスのみを起動
- データベースサーバ：
- セルにfs2.allbsd.orgを追加する
  - bos addhost fs.allbsd.org fs2.allbsd.org -localauth

# OpenAFSサーバ

2台目を追加するには？

- データベースサーバ、FSサーバ、どちらも複数持てる
  - 冗長性が自動的に確保され、使えるものが使われる
- バックアップ、レプリケーション構成は、
  FSサーバの追加時に指定する

# OpenAFSサーバ

kinitが必要ということは、cronとかはどうするの？

→ あらかじめ鍵を取り出しておいて kinitする

```
#!/bin/sh
/usr/bin/kinit -t /etc/cron.keytab cron/afs
/usr/local/bin/aklog -setpag

....

/usr/local/bin/unlog
/usr/bin/kdestroy
```

# まとめ

- 分散ファイルシステムの考え方
  → スケーラビリティが最重要課題
- OpenAFSをシングルボリュームで動かす設定方法
  → KerberosとDNSの設定も必要
  → bos, pts, vos などのコマンドを使う
- Kerberos鍵をkinitで取り出し、aklogしてアクセス権を得る。

- 次回は、より実用的な構成方法と、GlusterFSについて扱います。

# おしまい

- 質問はありますか？