



WEB 系システムに OpenSolaris を使うメリット

Name 瀧 康史

Title ジャストプレイヤー株式会社

代表取締役社長

Japan OpenSolaris User Group Leader

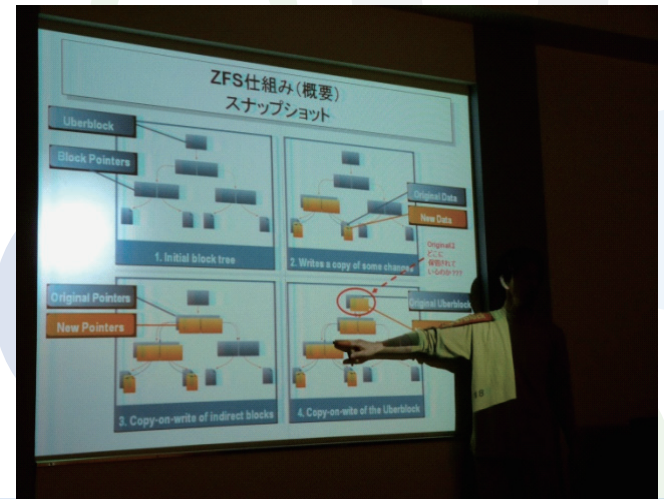
自己紹介

- 瀧 康史(たき やすし)
- ジャストプレイヤー株式会社 代表取締役社長
- 出身地
 - 静岡市 (旧清水市)
- 業務履歴
 - ライター業 (テクニカルライター)
 - S社 'X紙' (S社のパソコン向け雑誌) を中心に5年間。
 - S社にてS社用ゲーム機のゲームソフトを5年間制作。
 - ゲームディレクター、アシスタントプロデューサー、プランナーなどの雑用 (笑) 係
- 2001年、有限会社ジャストプレイヤーを起業
 - 現: ジャストプレイヤー株式会社



Japan OpenSolaris User Group

- 日本のOpenSolarisのコミュニティです。
- 日本ポータル(<http://jp.opensolaris.org>)があります。
- メーリングリスト(ug-jposug@opensolaris.org)で活発に意見交換されています。
- 月に一度、最終土曜日に、「東京OpenSolaris勉強会」を開催しています。
- IRC(<irc://irc.freenode.net/opensolaris-jp>)でも、眠らないディスカッションが……。
- 初心者大歓迎です。



ジャストプレイヤー株式会社について



～ソフトウェアに即時性を!～

<http://www.justplayer.co.jp/>

- JUSTPLAYER
 - 「我々はプレイヤーでありつづけたい」
 - (ソフトウェアが) すぐに遊べる (ミドルウェアの提供をしたい)
- WHEN YOU WANT IS WHEN YOU PLAY(欲しいときに遊ぶとき)
 - 「ソフトウェアに即時性を!」
 - ソフトウェアのオンデマンド配信が最初の事業

事業内容

- 事業内容
 - WEB(ホームページ)制作事業
 - WEB企画、デザイン制作
 - WEBシステム開発
 - SI/ソフトウェア開発事業
 - SaaS/ASP、ホスティング事業
 - <http://www.justplayer.ne.jp>
- その他
 - 第二種電気通信事業者届出 届出番号 C-18-1421
 - 静岡県中小企業新事業活動促進法（静岡県中小企業革新法）に基づく経営革新計画の承認（WikiベースのCMS、WikiPlus）



WEB会社としては若干珍しく、どちらかというとインフラに近い仕事が得意。また、それぞれの切り売りも得意。

Agenda

- IPSでインストールは簡単
- SMFで簡単起動と突然死の回避
- Zoneをつかって、サービスサーバを分離
- BARTで改竄検知
- lofs(loopback file system)で改竄ブロック
- *statやD-Traceでパフォーマンス解析
- SRMでリソースの隔離

ターゲット

本、セッションでは、次の方々をターゲットにしています。

- 普段、Linuxや(Free|net|Open|BSD)、などのOSでWEBサービスをしてる方
 - Apacheをよく使っている方
 - php、perl、python、ruby、javaなどでWEBアプリを使っている方
 - 等々
-
- 30分なので、細かなことを説明することはできないので、概要的なところをかいつままで説明していきます。
 - そのような方々に、“OpenSolaris、ちょっと使ってみたいかなあ〜”と思ってもらうことが今回の目的です。

WEB に何の言語を使っていますか？

皆さんはWEBに何の言語を使っていますか？

- php
- Perl
- Python
- Ruby
- Java
- などなど



DBは何を利用していますか？

バックヤードストレージ(DB、KVSなど)は、何を利用していますか？

- MySQL
- PostgreSQL
- memcached
- sqlite
- Oracle
- Hadoop

どれも、OpenSolarisで、簡単にインストール可能です。

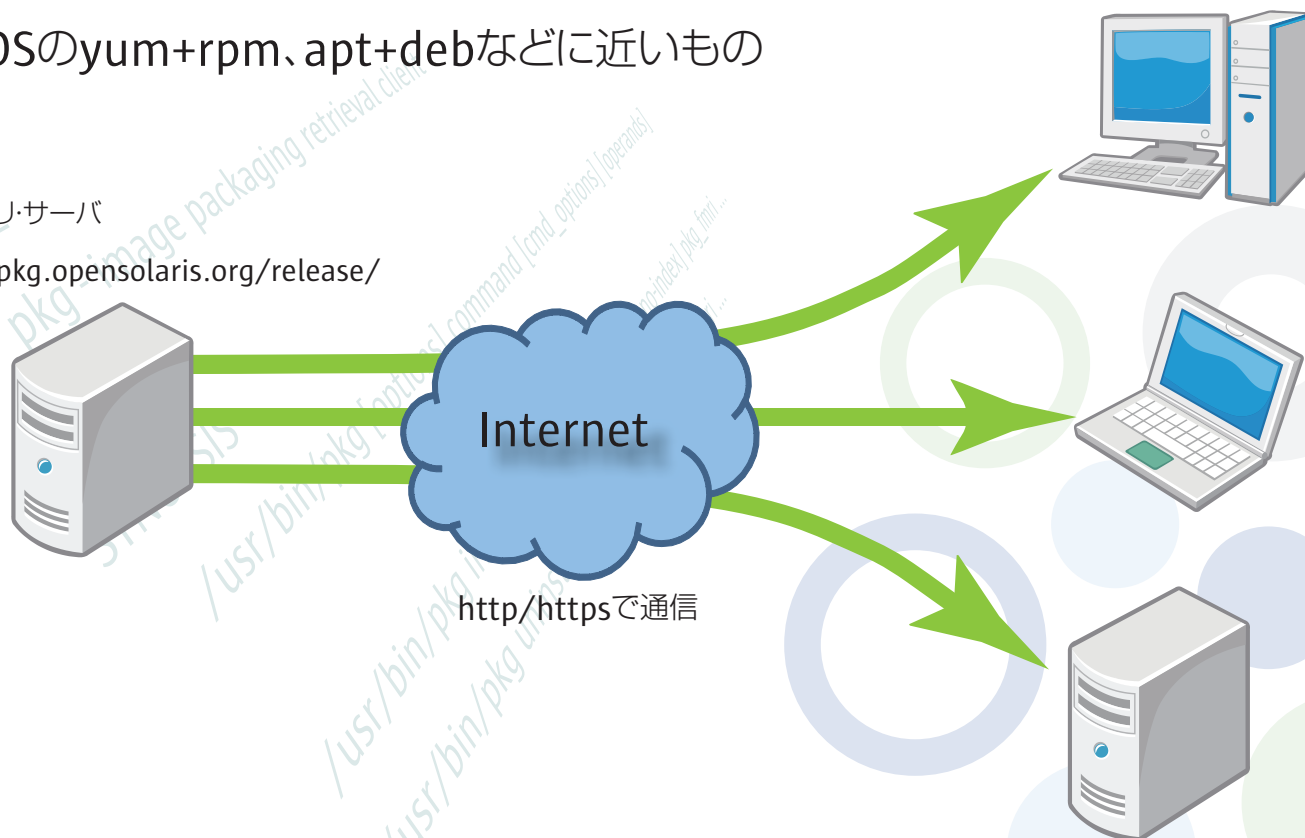
IPS でインストールは簡単

IPSは、OpenSolarisで利用されているパッケージングシステム

- ソフトウェアをInternet越しに簡単に配信・インストールできる
- Solaris10のSVR4パッケージシステム(pkgadd等)を代替する
- Linux系OSのyum+rpm、apt+debなどに近いもの

レポジトリ・サーバ

<http://pkg.opensolaris.org/release/>



AMP=Apache/MySQL/PHP

AMP=Apache、MySQL、PHPのインストール

```
pfexec pkg install -v amp
```

これは、と同じです。

```
pfexec pkg install -v SUNWmysql51 SUNWapch22m-security SUNWphp52 SUNWapch22m-dtrace SUNWphp52-mysql SUNWapch22m-fcgid SUNWapch22m-php52
```

含まれるパッケージ

```
SUNWmysql51@5.1.30-0.111  
SUNWapch22m-security@2.5.9-0.111  
SUNWphp52@5.2.9-0.111  
SUNWapch22@2.2.11-0.111  
SUNWapch22m-dtrace@0.3.1-0.111  
SUNWphp52-mysql@5.2.9-0.111  
SUNWapch22m-fcgid@2.2-0.111  
SUNWapch22m-php52@5.2.9-0.111
```

コマンドの対比

	IPS	pkgutil+pkgadd	pkg-get+pkgadd	yum+rpm	apt+dpkg
ファイル検索(インストール済/レポジトリ)	pkg search/pkg search -r	pkgchk -lp/なし	pkgchk -lp/なし	rpm -qf / yum provides	dpkg -S/apt-cache search
リスト取得(インストール済/レポジトリ)	pkg list/pkg list -a(リモートではなく全体)	pkginfo(pkgutil -c)/pkgutil -a	pkginfo(pkg-get -l) /pkg-get -D	rpm -qa/yum list	dpkg -l/apt-cache pkgnames
メタ情報取得(インストール済/レポジトリ)	pkg info/pkg info -r	pkginfo -l/	pkginfo -l/	rpm -qi/yum info	dpkg -s / apt-cache showpkg
内容の一覧(インストール済/レポジトリ)	pkg contents/pkg contents -r	pkgchk -l	pkgchk -l/	rpm -ql/?	dpkg -L/?
ベリファイ	pkg verify	pkgchk -v	pkgchk -v	rpm -V	debsums
インストール	pkg install	pkgutil -i (pkgadd)	pkg-get -i (pkgadd)	yum install	apt-get install
アップデート	pkg install	pkgutil -u	pkg-get -u	yum update/install	apt-get install
アンインストール	pkg uninstall	pkgutil -r (pkgrm)	pkg-get -r	yum uninstall	apt-get remove
環境全体のアップデート	pkg image-update / pkg install entire	pkgutil -U && pkgutil -u	pkg-get -U && pkg-get -u	yum upgrade	apt-get dist-upgrade

- IPSにはメタ情報検索系がなさそう。
- リポジトリとインストール済みのローカルで、コマンド体型が同じである。
- 他のパッケージシステムに比べて、メタ情報はキャッシュをあまりしない。

SMF で簡単起動

Apacheの起動方法

```
svcadm enable apache22
```

RHELなどの

```
service httpd start
```

や、レガシーな

```
/etc/init.d/httpd start
```

と、書き方が違うだけかな。と思うかもしれませんが……

- enable

- とにかくapache22は立ち上げていなさい!ということ。
- 再起動後も必ずapache22は立ち上げて!という意味です。
- 従来は、/etc/rc3.d/Sxxhttpdなどにリンクを張ってましたが、

- disable

- とにかく落としなさいということ。

SMF で簡単起動と突然死の回避

SMFの特徴は

- Daemonの管理ができる。
- もし、落ちていたら、すぐに起動し直す。
- 依存物も含めてenableにできる。
- 落ちてるかどうかをSMFが管理している。
- コンソールを切り離せている。

WEBサービスに向いているSMF

- WEBアプリケーションはステートレス
 - 落ちたらすぐに起動すれば、大体何とかかなります。

httpdが落ちるとき……

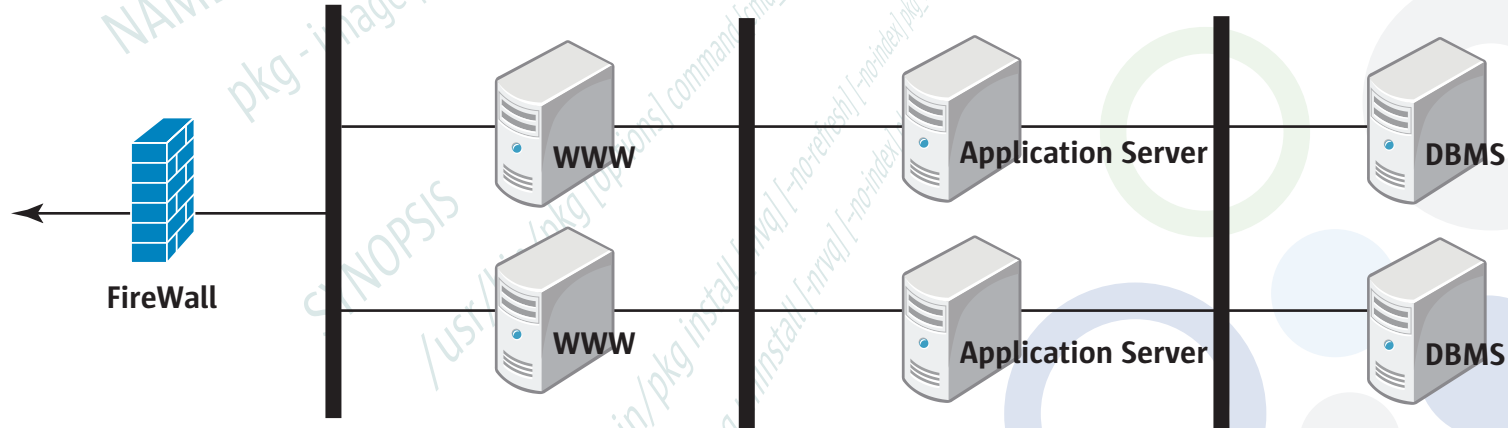
- ソフトウェアの問題
 - クリティカルなバグ。SEGV、プログラムがメモリリークしている。等。
- ハードウェアの問題。
 - メモリの破損、基盤のエラッタ、CPUが計算を間違える

一番汚染される場所とは？

割とシンプルな構成です。

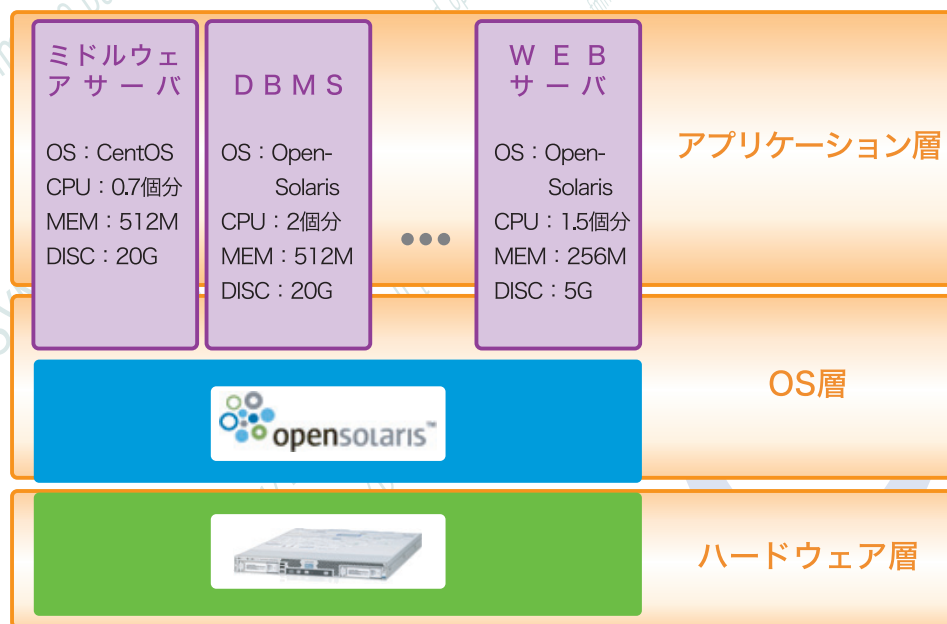
- WEB x 2
- Application Server x2
- DBMS x 2

このなかで、汚染されやすいサーバはどれですか？



Zone で、サービスサーバを分離

- Zoneを使うと、1台のマシンで複数の仮想サーバを動かすことができます。
- コンテナ型の仮想化システムは、リソース効率が良く、アプリケーションをコンテナの中にくまなく閉じこめることができます。
- 弊社のPhase2Serverはこのコンテナを、1つずつ貸すサービスです :)
- 仮想サーバから、下のOS層(GlobalZone)へは、アクセスできません。
- GlobalZoneから、Zoneの中身をファイルシステムをファイル単位でアクセス可能です。



BARTで改竄検知

BARTは改竄検知ソフトウェア

- ファイルのシグネチャの作成
- シグネチャと比較して異なったら報告

このようなソフトにも問題があって……

- コマンドインジェクションや、rootkitが仕込まれたりすると、改竄検知のシグネチャも怪しい。
- そもそも改竄検知が動かないなども。

GlobalZoneは、その上で動作するすべてのファイルシステムが見えるため、GlobalZoneで改竄検知のチェックプログラムを動かせば、Jail Breakが起きない限り大丈夫。

lofs で改竄ブロック

- そもそも、改竄検知がいない場合もあります。
- WEBサーバのDocumentRootがReadOnlyにする事ができます。
- lofsは、GlobalZoneの一部のファイルシステムを、Zoneに見せる仕組みです。
- この仕組みはRead Onlyで見せることができ、Zoneからはこのfsはroot権でもrwにマウントし直すことはできません。

SRM でリソースの隔離

1台のマシンで、次のようなサービスを動かすことはありませんか？

- WEBサーバ
- アプリケーションサーバ(phpなどでもいい)
- データベース
- メールサーバ

See: pooladm、psradm、resource_controls、projmod、poolbind

Container = Zone + SRM

ZONE+SRMで、Apache/PostgreSQLなどのリソースプールを分けることで、より安定的な動作をさせることも可能です。下記は、Zoneにリソースを割り当てた例。

SRMに比べると、だいぶ簡単にできます。

```
add rctl
set name=zone.cpu-cap
add value (priv=privileged,limit=150,action=deny)
end
add rctl
set name=zone.max-swap
add value (priv=privileged,limit=1073741824,action=deny)
end
add capped-memory
set physical=512M
end
```

上記は、CPUを150%以上使わせない。メモリ512MB、スワップ1GBのcap。