Monthly Research

SELinux 再入門
-仮想化・コンテナ編-

株式会社ＦＦＲＩ
http://www.ffri.jp

Ver 1.00.01

SELinux 再入門 -仮想化・コンテナ編-

- 仮想化技術（VM）とセキュリティ
  - VM特有の脅威モデル
  - libvirt(svirt)
  - Type Enforcement*1 を使ったホスト-ゲスト隔離
  - Multi Category Security*2を用いたゲスト間隔離

- Dockerとセキュリティ
  - Dockerとは
  - Dockerの脅威モデル
  - libcontainerによる隔離

- まとめ

*1, *2: SELinuxの基礎については、2014年06月のMonthly Researchを参照
http://www.ffri.jp/assets/files/monthly_research/MR201406_A%20Re-introduction%20to%20SELinux_JPN.pdf

はじめに

- VMを使ったOS仮想化環境においては、仮想化環境を提供するホストOSと仮想化環境であるゲストOSそれぞれの隔離と、ゲストOS間のセキュリティが重要である
  - LXC, dockerのようなコンテナ型OS仮想化においても同様

- 現状、多くの仮想化及びコンテナは実行にroot権限が必要なため、その“ゲストOS（コンテナ）の封じ込め”は非常に重要である

- 今回はlibvirtに統合されたSELinuxによるLinuxの仮想化システムの隔離と、dockerのlibcontainerを用いた隔離について紹介する

仮想化技術（VM）とセキュリティ

VM特有の脅威モデル

- ゲストOSがホストOSに対して攻撃を行うケース
  - ゲストOSはだいたいの場合特権ユーザで動作しているため、VMの脆弱性を用いて攻略されてしまう可能性がある

- ゲスト同士がお互いの情報を読み取ったり、改ざんを行うケース

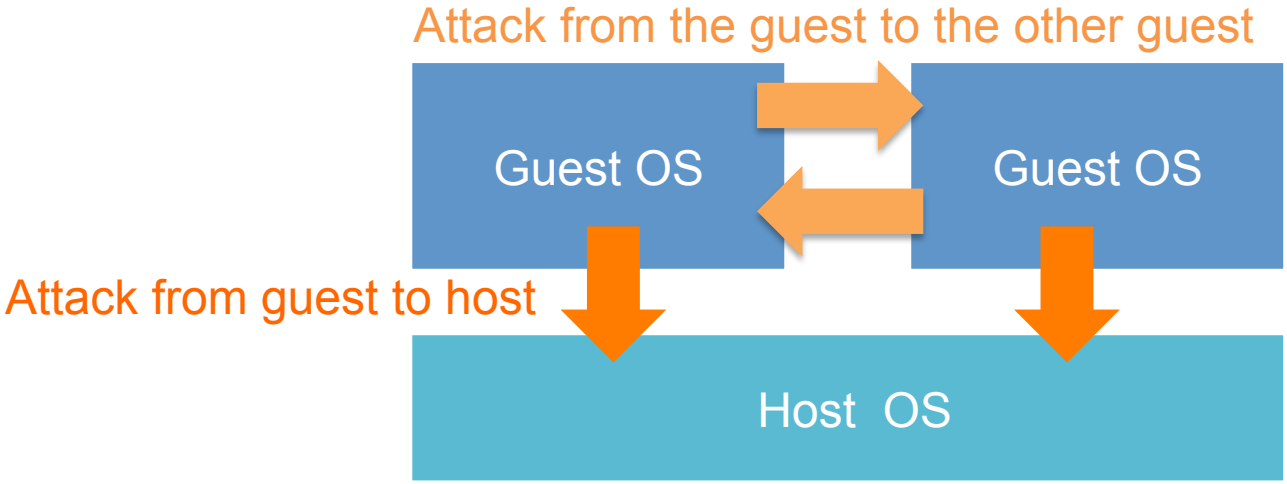


Fig 1. Threat model of virtualization system

libvirt(with svirt)

- 各種仮想化システムを統合するための管理フレームワーク
  - ハイパーバイザの違いを意識せずにVMを操作できるvirshもこのフレームワークの一部

- 各種仮想化システム（KVM、Xen、QEMU、VirtualBox、VMware 、Hyper-V、LXC、UMLなど）のプロセスのケーパビリティやSELinuxコンテキストを管理するsVirtが統合されている
  - ポリシーやセキュリティコンテキスト自体はreference policyで管理されている

Type Enforcement を使ったホスト-ゲスト隔離

- libvirtのSecurity Manager(svirt相当）は以下のアクセス制御機構を用いて、ゲストOSをホストOSから隔離する
  - selinux, apparmor, Unix DAC(uidベースのアクセス制御）

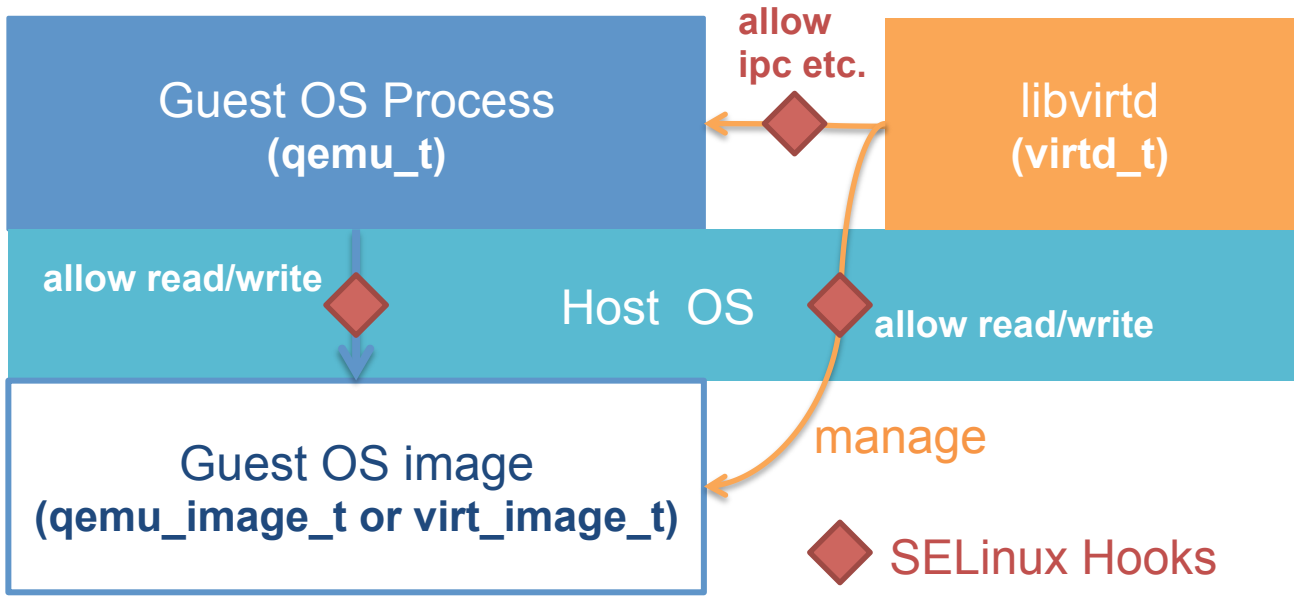


Fig 2. Host-Guest Isolation using SELinux

Multi Category Securityを用いたゲスト間隔離

- libvirtのSecurity ManagerはゲストOS・コンテナごとに固有のセキュリティラベルを割り当てる
- SELinuxの type enforcementは動的なtype割り当てが苦手なので、MCS（Multi Category Security）を使ってカテゴリを分ける

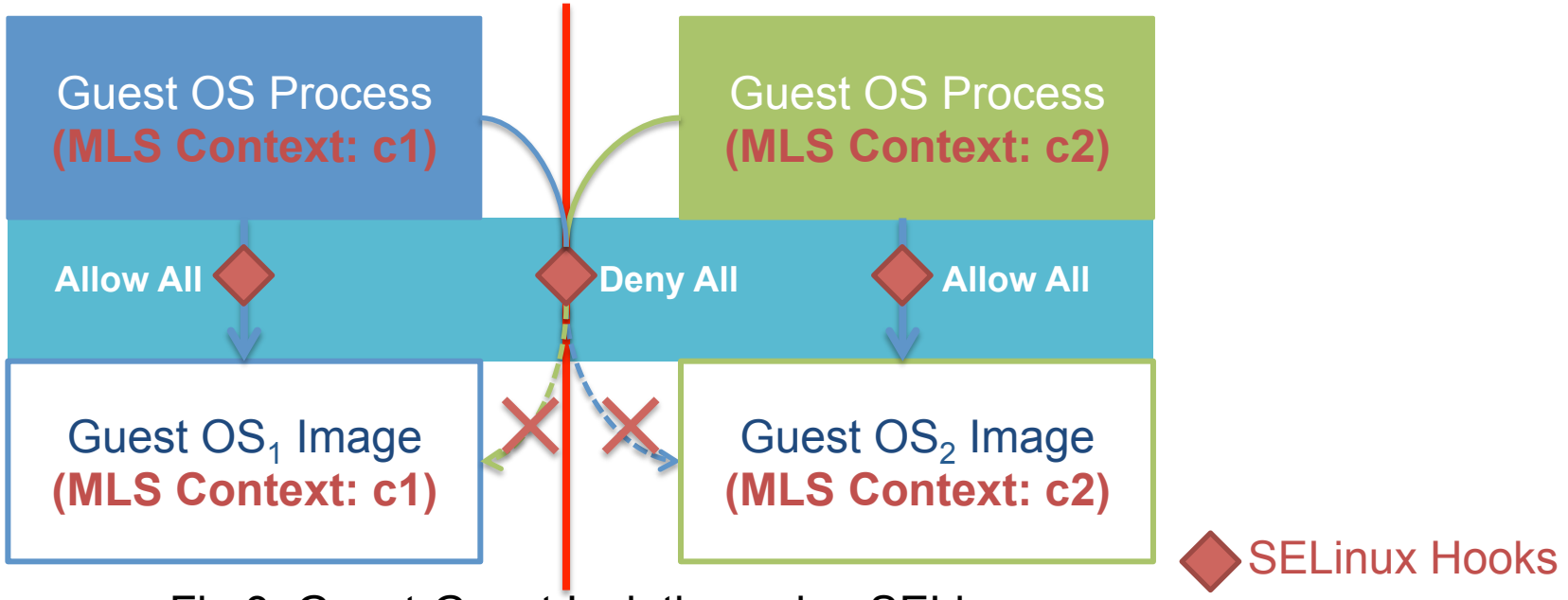


Fig 3. Guest-Guest Isolation using SELinux

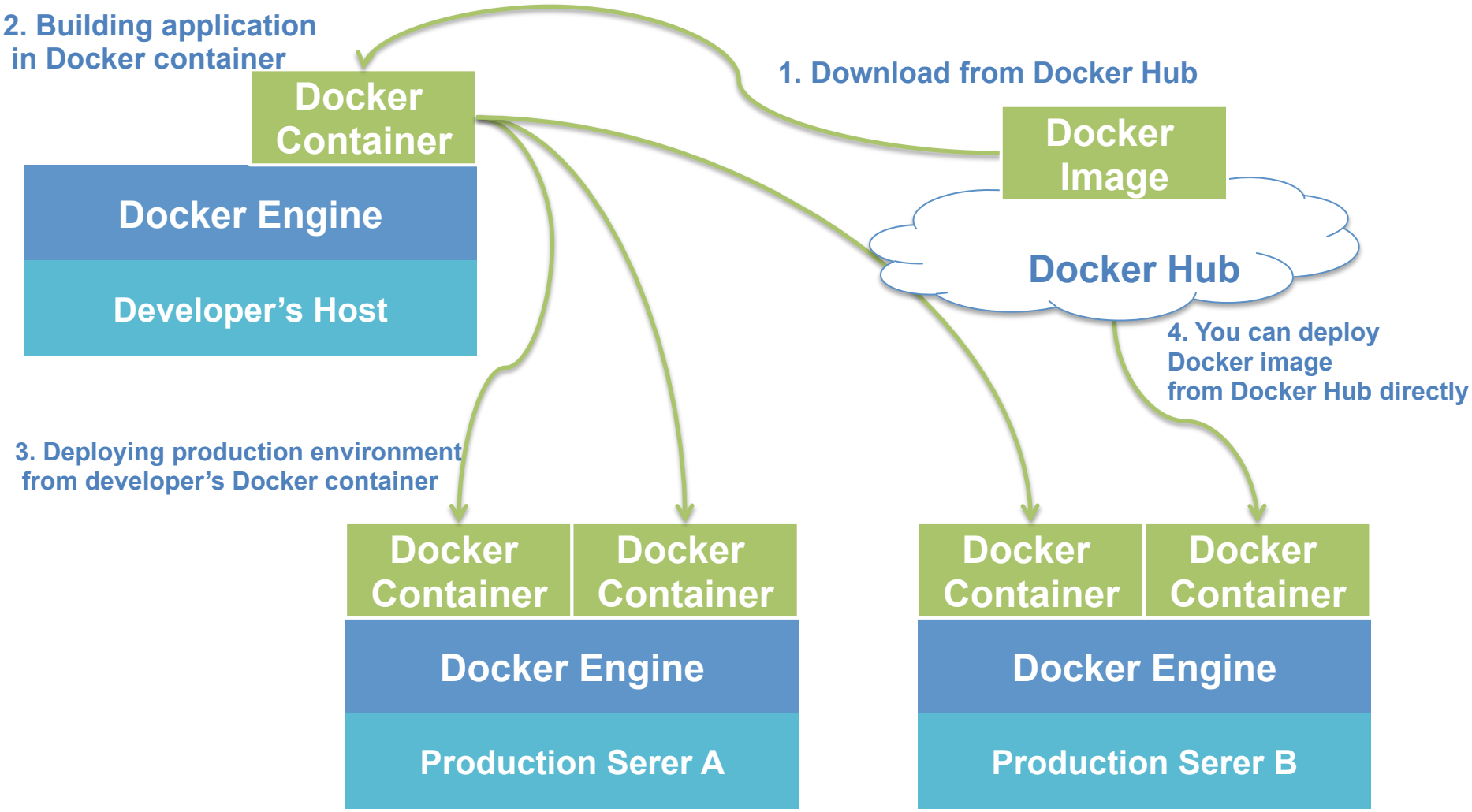
DOCKERとセキュリティ

Dockerとは

- アプリケーションのデプロイにフォーカスした、コンテナ型OS仮想化と差分ディスク管理によるコンテナ管理基盤
  - Docker Engine
    - OS上で動作するマネージャ
  - Docker Hub
    - Web上でDockerイメージを管理、配信するプラットフォーム（Githubのコンテナイメージ版）

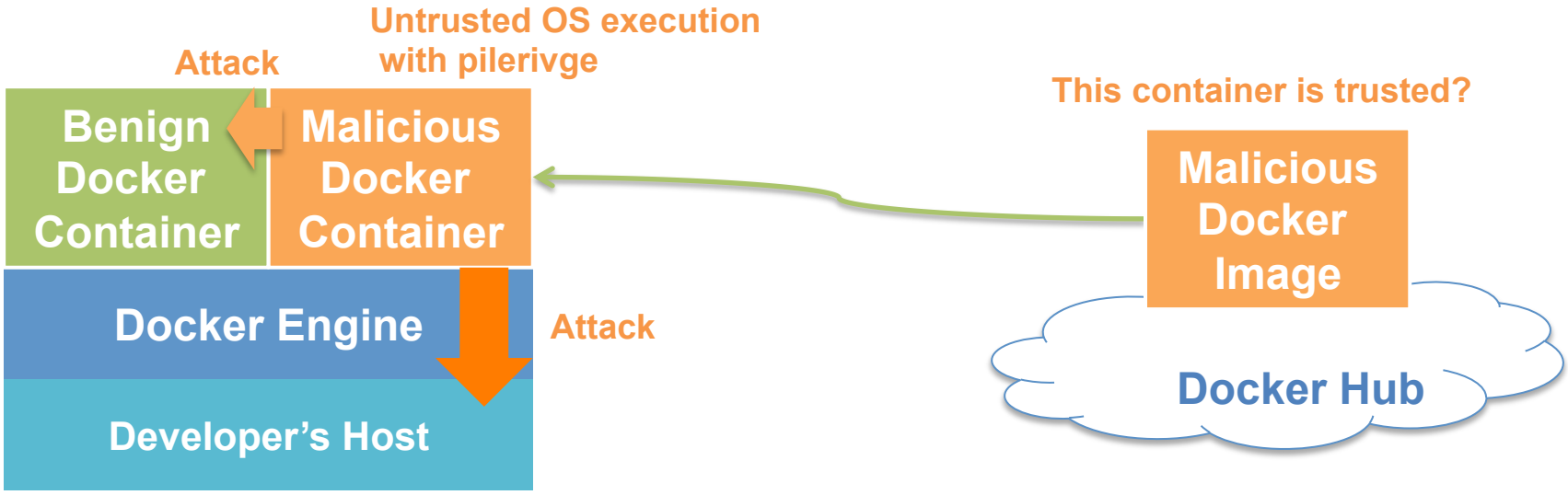
- 元は米DotCloud（現docker）が開発したPaaS基盤
  - Amazon, Google, Microsoftが対応を表明するなど、非常にホットな技術

Scalable Deployment Cycle with Docker



Dockerの脅威モデル

- Docker repositoryからOSイメージをダウンロードし、コンテナとして実行することは、例えるなら信頼できないプログラムをダウンロードし、ローカルで実行するようなものと同じともいえる
  - public imageに悪意がないと言い切ることは難しい



libcontainerによる隔離

- Dockerは従来lxcとその管理コマンドlibvirt-lxcを用いて隔離を実現していたが、0.9からlibcontainerを用いた隔離に変更した
  - コンテナ-ホスト間の隔離は、Linux capabilities とcgroup、MAC(SELinux, AppArmor)を用いて行われる
  - コンテナ間の隔離は、MAC(SELinux, AppArmor)を用いて実現している
    - SELinuxを使う場合、MCSセキュリティコンテキストを動的に生成し、割り当てる

- <~0.9で実現される隔離と0.9以降における隔離はほぼ同一
  - libvirtへの依存を減らし、他のディストリビューションやOSへ適用しやすくしたものと考えられる

まとめ

- 仮想化やコンテナ技術を使う場合、2つの視点で隔離を行う必要がある
  - ホスト-ゲスト間の隔離
    - ホストOSを守れなければ、他のゲストOSも危ない
  - ゲスト間の隔離
    - ゲストOSから他のゲストOSを侵害させない
- libvirtは洗練された仮想化基盤管理アプリケーションであり、SELinuxやAppArmorも統合されている
- dockerのようなコンテナ型仮想化においても、脅威モデルはVMのセキュリティと変わらない

参考文献

- 第3章 sVirt
  https://access.redhat.com/documentation/ja-JP/Red_Hat_Enterprise_Linux/6/html/Virtualization_Administration_Guide/chap-sVirt.html
- Docker and SELinux
  http://www.projectatomic.io/docs/docker-and-selinux/
- DOCKER 0.9: INTRODUCING EXECUTION DRIVERS AND LIBCONTAINER
  https://blog.docker.com/2014/03/docker-0-9-introducing-execution-drivers-and-libcontainer/
- Introducing a *Super* Privileged Container Concept
  http://developerblog.redhat.com/2014/11/06/introducing-a-super-privileged-container-concept/

Contact Information

E-Mail : research—feedback@ffri.jp
Twitter : @FFRI_Research