WithSecure Linux セキュリティ 11.x をインストールする前のチェックリスト

F-Secure Linux セキュリティ 11.x をインストールする前のチェックリスト

一部の Linux ディストリビューションでは、F-Secure Linux Security 製品をインストールする前に特定のソフトウェアパッケージのインストールや特殊な設定が必要になる場合があります。この記事では、最も一般的な構成と関連するソリューションについて説明します。

Prelink を使うディストリビューション

Prelink はバイナリの起動時間を短縮できますが、本製品の完全性検査と競合します。

prelink を無効にするには、ご使用の OS で設定ファイル (たとえば`/etc/sysconfig/prelink`) を探し、次の行を `PRELINKING=yes` から `PRELINKING=no` に変更して`/etc/cron.daily/prelink`、製品をインストールします。

cron から Prelink の自動実行を無効にすることを推奨します。ディストリビューションのなかにはダイナミックライブラリを使用するバイナリの起動時間を短縮するために、prelink を cron から定期的に実行するものがあります。Prelink はディスク上のバイナリとダイナミックライブラリを変更するため、システムファイルの変更を検出する完全性検査と競合します。

F-Secure Linux Security をすでにインストールしている場合、以下の手順を実行します。

1. `/opt/f-secure/fsav/bin/fsims on` をコマンドラインから実行して、ソフトウェアインストールモードを有効にします。ソフトウェアインストールモードでは、製品はシステムファイルへの変更を許可します。
2. `/etc/sysconfig/prelink` を編集します。`PRELINKING=yes` を `PRELINKING=no` に変更します。
3. `/etc/cron.daily/prelink` を実行します。
4. `/opt/f-secure/fsav/bin/fsims off` をコマンドラインから実行して、ソフトウェアインストールモードを無効にします。

ソフトウェアインストールモードを無効にした時点で、システムファイルの状態が完全性検査のベースラインに保存されます。

Prelink を使用する場合、使用前にソフトウェアインストールモードを有効にし、使用後にソフトウェアインストールモードを無効にする必要があります。これによって、Prelink はシステムファイルに変更を加えるようになります。例 :

```
# /opt/f-secure/fsav/bin/fsims on # prelink -a # /opt/f-secure/fsav/bin/fsims off
```

注: この処理は簡単に自動化できません。ソフトウェアインストールモードを無効にすることによって、新しいベースラインが作成され、その際に管理者のパスワードが確認されます。

インストール前の要件

製品をインストールする前に、以下のパッケージをインストールする必要があります。64 ビットの環境では、32 ビットランタイムサポートをインストールする前に、Multiarch サポートを有効にする必要があります。Dazuko カーネルドライバを使用するディストリビューションでは、カーネルヘッダとコンパイラツールもインストールする必要があります。

カーネルドライバをコンパイルするために、現在使用している kernel、kernel-devel、kernel-headers のパッケージバージョンを一致させる必要があります。

CentOS/RHEL 6 (32 ビット)

```
yum install gcc glibc-devel glibc-headers kernel-devel make pam patch perl
```

Debian 7 (32 ビット)

```
sudo apt-get install gcc libc6-dev libpam-modules linux-headers-$(uname -r) make patch perl rpm
```

Debian 8/9 (32 ビット)

```
sudo apt-get install rpm pam perl
```

Ubuntu 12.04/12.04.1/12.04.2 (32 ビット)

```
sudo apt-get install gcc linux-headers-$(uname -r) perl rpm
```

Ubuntu 12.04.3/12.04.4/12.04.5 (32 ビット)

```
sudo apt-get install rpm
```

SUSE Linux Enterprise Server 11 (32 ビット)

```
sudo zypper in gcc kernel-default-devel make patch perl
```

Oracle Linux 6 RHCK (32 ビット)

```
yum install gcc glibc-devel kernel-devel make patch perl
```

Amazon Linux 2017.03 / 2017.09 / 2018.3 (64 ビット)

```
yum install libstdc++44.i686 pam.i686
```

CentOS/RHEL 6 (64 ビット)

```
yum install gcc glibc-devel glibc-headers glibc.i686 glibc.x86_64 kernel-devel libstdc++.i686
libstdc++.x86_64 make pam.i686 pam.x86_64 patch perl zlib.i686 zlib.x86_64
```

CentOS/RHEL 7 (64 ビット)

```
yum install glibc.i686 glibc.x86_64 libstdc++.i686 libstdc++.x86_64 pam.i686 pam.x86_64 perl
zlib.i686 zlib.x86_64
```

Debian 7 (64 ビット)

1. Multiarch サポートを有効にするには

   ```
   dpkg --add-architecture i386 apt-get update
   ```

2. 次のパッケージをインストールします。

```
sudo apt-get install gcc libc6-dev libpam-modules:i386 libstdc++6:i386 linux-headers-$(uname -r) make patch perl rpm zlib1g:i386
```

Debian 8/9 (64 ビット)

1. Multiarch サポートを有効にするには
   dpkg --**add**-architecture i386 apt-get update
2. 次のパッケージをインストールします。
   ```
   sudo apt-get install libpam-modules:i386 libstdc++6:i386 perl rpm zlib1g:i386
   ```

Ubuntu 12.04/12.04.1/12.04.2 (64 ビット)

```
sudo apt-get install gcc libpam-modules:i386 libstdc++6:i386 linux-headers-$(uname -r) perl rpm zlib1g:i386
```

Ubuntu 12.04.3/12.04.4/12.04.5 (64 ビット)

```
sudo apt-get install libpam-modules:i386 libstdc++6:i386 rpm zlib1g:i386
```

Ubuntu 14.04/16.04/18.04 (64 ビット)

```
sudo apt-get install libc6-dev:i386 libpam-modules:i386 libstdc++6:i386 rpm zlib1g:i386
```

SUSE Linux Enterprise Server 11-SP1/11-SP2/11-SP3 (64 ビット)

```
sudo zypper in gcc kernel-default-devel libgcc43-32bit libstdc++43-32bit make pam-modules-32bit patch perl
```

SUSE Linux Enterprise Server 11-SP4 (64 ビット)

```
sudo zypper in gcc kernel-default-devel libgcc_s1-32bit libstdc++6-32bit make pam-modules-32bit patch perl
```

SUSE Linux Enterprise Server 12 (64 ビット)

```
sudo zypper in libstdc++6-32bit libz1-32bit pam-32bit
```

Oracle Linux 6 RHCK (64 ビット)

```
yum install gcc glibc-devel glibc-devel.i686 kernel-devel libstdc++.i686 make pam.i686 patch perl zlib.i686
```

Oracle Linux 7 UEK (64 ビット)

```
yum install libstdc++.i686 pam.i686 zlib.i686
```

Linux Security を初期化する

製品をインストールする前にパッケージの依存関係が見つからなかった場合、パッケージをインストールした後に次のコマンドを実行してすべての F-Secure モジュールを正しく初期化します。
`/etc/init.d/fsma restart`
Linux Security カーネルインターセプターをコンパイルできない場合、次のコマンドを実行してください。`/opt/f-secure/fsav/bin/fsav-compile-drivers`
`fsav-compile-drivers`
は fsma restart も実行します。