Dr.WEB®
Anti-virus
for UNIX Internet gateways

Defend what you create

管理者マニュアル

Dr.We® for Unix Internet gateways
バージョン 6.0.1
管理者マニュアル
27.10.2011

ロシア本社
2-12A, 3rd str. Yamskogo polya
Moscow, Russia
125124

ウェブサイト www.drweb.com
電話 +7 (495) 789-45-87

地方支店、オフィスに関する情報は、弊社のオフィシャルサイトにあります。

Doctor Web

弊社はマルウェアおよび迷惑メールに対する効率的な保護を
提供する Dr.WebR 情報セキュリティソリューションの
開発および販売を行っています。

個人ユーザから政府機関、また中小企業から国際的な企業まで、
世界中のあらゆる地域に弊社のお客様は広がっています。

Dr.Web アンチウイルスソリューションは 1992年以来、
卓越したマルウェアの検出能力と国際的な情報
セキュリティ基準への適合で良く知られています。
Dr.Web ソリューションには政府による認証や表彰が
何度も与えられていること、また弊社製品のユーザが
世界中に広がっていることは、弊社製品に対する
皆さまからの絶大な信頼の証しだと自負しています。

**弊社の全てのお客さまからの多大なるご支援と
ご貢献に心より感謝いたします。**

目次

はじめに

Dr.Web for UNIX Internet gatewaysをご購入いただきありがとうございます。

本書は、**Dr.Web for UNIX Internet gateways**のインストールおよびアンインストール、基本設定、詳細な設定例などについて説明している管理者用ガイドです。

Dr.Web for UNIX Internet gateways は、様々なUnixベースのOS上で動作し、企業のインターネットゲートウェイとしてウイルス対策の機能を提供します。

- **Dr.We® for Unix Internet gateways** for Linux
- **Dr.We® for Unix Internet gateways** for FreeBSD
- **Dr.We® for Unix Internet gateways** for Solaris

本製品は以下の3つの特徴を持っています。

- HTTPおよびFTPトラフィックのウイルススキャン
- URLフィルタリング
- **Dr.Web Enterprise Security Suite Control Center** Webによる集中管理

Dr.Web for UNIX Internet gatewaysは、3つの主要コンポーネントによって構成されています。

- **Dr.Web Scanner** ローカルマシン上のウィルス検査を行うコマンドラインスキャナです。
- **Dr.Web Daemon** フィルタからの要求により、ファイルのウイルス検査を行うデーモンです。
- **Dr.Web ICAP** デーモンにウィルス検査の依頼をするフィルタで、ICAPプロトコルを使用するアプリケーションと連携して動作します。

本書は以下の内容について説明しています。

- 製品概要
- インストール
- スタートアップ

- **Dr.Web Updater**の使用方法
- **Dr.Web Control Agent**の使用方法
- **Dr.Web Monitor**の使用方法
- **Dr.Web Scanner**の使用方法
- **Dr.Web Daemon**の使用方法
- **Dr.Web ICAP**の使用方法
- **Dr.Web console for UNIX Internet gateways**の使用方法

表記規則

本書では、以下の文字・記号を使用しています。

文字・記号	意味
太字	グラフィカルユーザインターフェース(GUI)の要素の名称や本書のとおり正確に入力する必要のある入力例
緑色の太字	**Doctor Web** 製品またはコンポーネントの名称
緑色で下線付きの文字	本書の他のページや他のWebページへのリンク
固定幅フォント	コマンドラインの入力例、出力例
イタリック体	ユーザが提供しなければならない情報を表すプレースホルダ。コマンドラインの入力例がイタリック体の場合は、パラメタ値を示します。
大太字	キーボードのキー名称
プラス記号 ('+')	キーの同時押し(例: ALT+F1 は、ALTキーとF1キーを同時に押すことを意味します。)
感嘆符	重要な注釈、またはエラーなどを引き起こす可能性のある状況に関する警告

ソフトウェアのコンポーネントがインストールされるディレクトリを定義するために `%bin_dir`, `%etc_dir`, `%var_dir` の表記を使用しています。

使用するOSごとに、それぞれ以下のディレクトリを指します。

Linux, Solaris:

```
%bin_dir = /opt/drweb/
%etc_dir = /etc/drweb/
%var_dir = /var/drweb/
```

FreeBSD:

```
%bin_dir = /usr/local/drweb/
%etc_dir = /usr/local/etc/drweb/
%var_dir = /var/drweb/
```

システム要件

Dr.Web for Unix Internet gatewaysは、以下の要件を満たすシステムで使用することができます。

対応OS:

- Linux（カーネル2.6 以降）
- FreeBSD 6以降（Intel x86）
- Solaris 10（Intel x86）

32bit/64bit（x86_64）環境に対応していますが、64bit環境では32bitアプリケーションの動作をサポートする必要があります。

必須要件:

- **Dr.Web Daemon**（drwebd）4.33以降
- Squid 3.0.STABLE1以降、またはSafeSquid 3.0以降
- Solaris 10の場合、libstdc++.so.4 ライブラリが必要

  libstdc++.so.4 は、gcc-3.3.2 または libgcc-3.3.と共にインストールされます。libstdc++.so.4が存在するにもかかわらず、drweb-icapdの起動時にlibstdc++.so.4の確認ができない場合、以下のようなコマンドで環境変数（LD_LIBRARY_PATH）を変更してください。

```
# export LD_LIBRARY_PATH="/usr/local/lib"
```

また、以下のパッケージがインストールされていることが必要です。

- libglade2
- libgtk2
- libstdc++6
- base64
- unzip
- crond

グラフィカルインターフェース(GUI)によるインストールを行うには、この他にX Window Systemの環境が必要です。X Window Systemが利用できない場合は、インストール用スクリプトを使用してインストールを行います。

必要メモリとディスク:

- 512MB以上のメモリ
- 205MB以上の空きディスク

パッケージファイル

Dr.Web for Unix Internet gatewaysは、デフォルトで %bin_dir, %etc_dir, %var_dir ディレクトリにインストールされます。

- %bin_dir - **Dr.Web for Unix Internet gateways**の実行モジュールおよび**Dr.Web Updater** (update.pl)
- %bin_dir/lib/ - ウイルス検査エンジンのライブラリ(drweb32.dll)
- %bin_dir/agent/ - **Dr.Web Agent**の追加設定ファイル
- %etc_dir/monitor/ - **Dr.Web Monitor**の追加設定ファイル
- %var_dir/bases/*.vdb - ウイルス定義ファイル
- %etc_dir - **Dr.Web for Unix Internet gateways**の設定ファイル: drweb32.ini, agent.conf, monitor.conf, drweb-icapd.ini, drwebd.enable, drweb-monitor.enable

- %bin_dir/lib/ru_scanner.dwl – **Dr.Web Scanner** の言語ファイル
- %bin_dir/doc/ – ドキュメントファイル
- %bin_dir/web/ – **Dr.Web for Unix Internet gateways** のWebインターフェースで使用するWebminパッケージ(drweb-icapd-web.wbm.gz)
- %var_dir/infected/ – 感染ファイルまたは、感染が疑われるファイルを隔離するためのディレクトリ

設定ファイル

Dr.Web for Unix Internet gatewaysの設定は、設定ファイルに定義されています。設定ファイルはテキストファイルで以下のように記述されています。(テキストエディタを使用することで、設定ファイルを編集することができます。)

```
--- beginning of the file ---
[Section 1 name]
Parameter1 = value1, ..., valueK
...
ParameterM = value1, ..., valueK
...
[Section X name]
Parameter1 = value1, ..., valueK
...
ParameterY = value1, ..., valueK
--- end of the file ---
```

";" または "#" で始まる行は、コメント行です。コメントアウトされた行は、設定ファイルのパラメータを読み込むときにスキップされます。コメント行、または値が指定されていないパラメータの場合は、ハードコード化されたデフォルト値が使用されます。

不正なパラメータが指定されている場合、**Dr.Web for Unix Internet gateways**はエラーメッセージを出力して終了します。設定ファイルに未知のパラ

メータを見つけると、**Dr.Web for Unix Internet gateways**はログファイルに警告を出力し続けます。

パラメータの指定は、以下のようにカンマ区切り、または複数行で複数の値を指定することができます。複数の値を指定できるパラメータについては、個々のパラメータの説明で記載があります。

例:

カンマ区切り:

```
Names = XXXXX, YYYYY
```

複数行:

```
Names = XXXXX
Names = YYYYY
```

本書では、全てのパラメータについて、以下の内容を説明しています。

- `ParameterName = {xxxx}` パラメータの名称と指定可能な値
- パラメータの説明
- 複数の値を指定できる場合の説明
- 初期値

パラメータにはいくつかの種類があり、それぞれ以下の値を指定することができます。

- `numerical value` – 0以上の整数
- `time` – 時間を示す0以上の整数

  `s` 秒
  `m` 分
  `h` 時
  `d` 日

  例: 30s, 15m
- `size` – ファイルサイズなどを示す0以上の整数

  `b` バイト
  `k` キロバイト

m　メガバイト
g　ギガバイト

例: 20b, 15k

大文字と小文字は区別しません。単位が省略された場合はバイトで値が設定されます。

- permissions – ファイルとディレクトリに与えられるアクセス権を示す数値

  例: 755 (-rwxr-xr-x), 644 (-rw-r--r--)
- path to file/directory – ファイルまたは、ディレクトリへのパス
- actions – 実行される動作、処理（パラメータごとに実行可能な動作が異なるため、それぞれのパラメータで説明があります。）
- address – **Dr.Web for Unix Internet gateways** コンポーネントと外部パッケージのソケット
  - inet – TCPソケットをPORT@HOST_NAMEの形式で指定します。HOST_NAMEは、ホスト名・IPアドレスのどちらでも指定できます。

    例:

    ```
    Address = inet:3003@127.0.0.1
    ```
  - local – ローカルのUNIXソケットをソケットファイルへのパスで指定します。

    例:

    ```
    Address = local:/var/drweb/run/.drwebd
    ```
  - PID – PIDファイル（指定可能な場合、個々のパラメータの指定で記載があります。）
- text value – テキスト文字
- string – カンマ区切りのテキスト文字で指定するテキスト文字
- value – 値

インストールとアンインストール

本章では、**Dr.Web for Unix Internet gateways**のインストールとアンインストールについて説明しています。記載されている手順は、root権限で実行する必要があります。

Dr.Web for Unix Internet gateways のインストールとアンインストールには、ESP Package Manager(EPM)形式のパッケージを使用することができます。X Window Systemが利用できる環境では、GUIによる簡単なインストールとアンインストールが可能です。尚、X Window Systemが利用できない環境であっても、インストール用スクリプトを使用したインストールとアンインストールが可能です。また、インストール後のサービス設定・起動を行うための設定用のスクリプトも用意されています。

- EPMパッケージによるGUIインストール、アンインストール
- コンソール環境でのインストール、アンインストール(インストール・アンインストール用スクリプト)

distribution package for UNIX systemsからのインストール

Dr.Web for Unix Internet gatewaysは、自己抽出パッケージとして提供されます。

```
drweb-internet-gateways_[version number]~
[OS name].run
```

パッケージには、以下のコンポーネントが含まれています。

- `drweb-common`: 設定ファイル(`drweb32.ini`)、ライブラリ、ドキュメント、ディレクトリ構造。インストールにより、drwebユーザとdrwebグループが作成されます。
- `drweb-bases`: ウイルス検査エンジン、ウイルス定義ファイル。`drweb-common`パッケージがインストールされている必要があります。
- `drweb-libs`: すべてのコンポーネントに必要な共通ライブラリ
- `drweb-epm6.0.0-libs`: GUIインストーラ・アンインストーラのライ

ブラリ。drweb-libsパッケージがインストールされている必要があります。

- drweb-epm6.0.0-uninst: GUIアンインストーラに必要なファイル。drweb-epm6.0.0-libsパッケージがインストールされている必要があります。
- drweb-boost144: **Dr.Web Agent**と**Dr.Web Monitor**の共通ライブラリ。drweb-libsパッケージがインストールされている必要があります。
- drweb-updater: ウイルス検査エンジンとウイルス定義ファイルのアップデートユーティリティ。drweb-common, drweb-libsパッケージがインストールされている必要があります。
- drweb-agent: **Dr.Web Agent**の実行ファイル、ドキュメント。drweb-common, drweb-boost144パッケージがインストールされている必要があります。
- drweb-monitor: **Dr.Web Monitor**の実行ファイル、ドキュメント。drweb-common, drweb-boost144パッケージがインストールされている必要があります。
- drweb-daemon: **Dr.Web Daemon**の実行ファイル、ドキュメント。drweb-bases, drweb-libsパッケージがインストールされている必要があります。
- drweb-scanner: **Dr.Web Scanner**の実行ファイル、ドキュメント。drweb-bases, drweb-libsパッケージがインストールされている必要があります。
- drweb-icapd: **Dr.Web ICAP**の実行ファイルとドキュメント。drweb-common, drweb-icapd-dws, drweb-libsパッケージがインストールされている必要があります。
- drweb-icapd-dws: URLフィルタリングのブラックリスト、ホワイトリスト。drweb-commonパッケージがインストールされている必要があります。
- drweb-icapd-web: **Dr.Web for Unix Internet gateways**のWebユーザインターフェース
- drweb-internet-gateways-doc: **Dr.Web for Unix Internet gateways**のドキュメント

64bit版のパッケージには、drweb-libsとdrweb-libs32が含まれています。(64bit用と32bit用)

Dr.Web for Unix Internet gatewaysのすべてのコンポーネントを自動的にインストールするためにコンソール(CLI)または、GUIベースのシェルを使用することができます。以下のようなコマンドで、インストールパッケージに実行権を与えてください。

```
# chmod +x drweb-internet-gateways_
[version number]~[OS name].run
```

パッケージを実行します。

```
# ./drweb-internet-gateways_[version number]~
[OS name].run
```

`drweb-internet-gateways_[version number]~[OS name]`の形式でディレクトリが作成され、GUIインストーラの初期化が行われます。root権限がない場合は、rootのパスワードを要求されます。

GUIインストーラが起動しない場合は、コンソール(CLI)のインストーラが初期化されます。

インストールを開始せずに、パッケージの抽出のみを行う場合は、以下のように `--noexec`パラメータを指定します。

```
# ./drweb-internet-gateways_[version number]~
[OS name].run --noexec
```

パッケージの抽出を行ったあとに以下のコマンドを実行することで、GUIインストーラの初期化とインストールが行えます。

```
# drweb-internet-gateways_[version number]~
[OS name]/install.sh
```

コンソール(CLI)のインストーラを初期化する場合は、以下のコマンドを実行します。

```
# drweb-internet-gateways_[version number]~
[OS name]/setup.sh
```

インストールによって以下の処理が行われます。

- オリジナルの設定ファイルが%etc_dir/software/conf/ディレクトリに保存されます。([configuration_file_name].N)
- 各種プログラムファイル、ディレクトリが配置されます。
- 既に同じ名前のプログラムファイルが存在する場合は、コピー([file_name].O)を保存した上で、新しいファイルで上書きをします。(旧バージョンのアンインストールが適切に行われなかった場合など)

distribution package for UNIX systemsのアンインストール

Dr.Web for Unix Internet gatewaysをアンインストールする場合、以下のコマンドを実行します。

```
# drweb-internet-gateways_
[version number]~[OS name]/remove.sh
```

root権限がない場合は、rootのパスワードを要求されます。

GUIアンインストーラが起動しない場合は、コンソール(CLI)のアンインストーラが起動します。

アンインストール後、drwebユーザとdrwebグループを削除することができます。

アンインストールによって以下の処理が行われます。

- オリジナルの設定ファイルを%etc_dir/software/conf/ディレクトリから削除します。
- ユーザによって設定ファイルを編集していた場合は、ファイルを残します。
- **Dr.Web**のファイルが削除されます。インストール時に[file_name].O が作成されていた場合、インストール前の名前で復元されます。
- ライセンスキーとログファイルは対応するディレクトリに残されます。

distribution package for UNIX systemsのアップグレード

Dr.Web for Unix Internet gatewaysのアップグレードは、既存のパッケージをアンインストールしてから最新バージョンのパッケージを新たにインストールすることで行います。

アンインストール時、ライセンスキーとログファイル、編集済みの設定ファイルは対応するディレクトリに残されます。

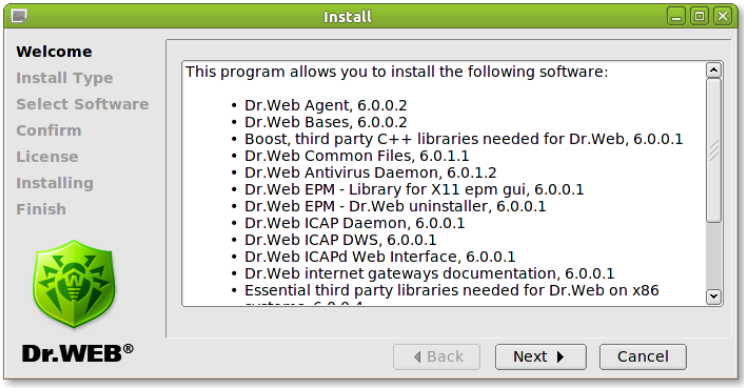
GUIインストーラによるインストール

1. 以下のコマンドを実行し、インストールを開始します。

```
# drweb-internet-gateways_
[version number]~[OS name]/install.sh
```

GUIインストールの画面が表示されます。

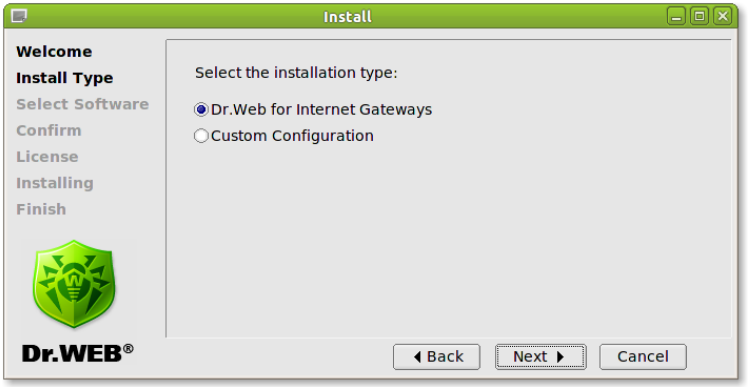
図 1. GUIインストール起動画面



"Next" ボタンを押して次に進みます。インストールを終了する場合は、**"Cancel"** ボタンを押します。

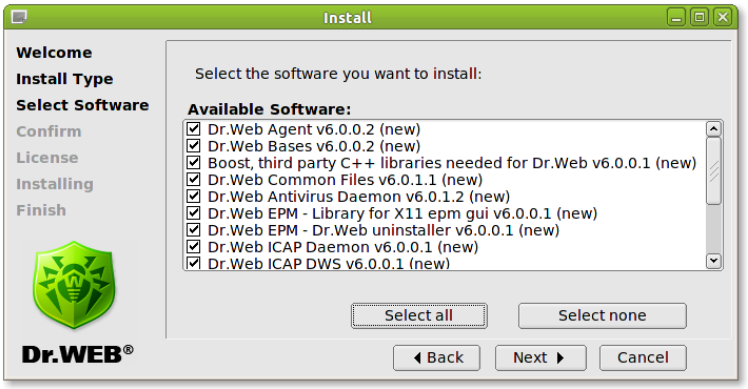
2. インストール種別の選択画面が表示されます。通常は、**"Dr.Web for Internet Gateways"** を選択し、**"Next"** ボタンを押してください。

図 2. インストール種別の選択画面



既にDr.Web製品がインストールされている場合など、必要に応じて **"Custom Configuration"** を選択し、インストールするコンポーネントを個別に選択することも可能です。

図 3. コンポーネントの選択画面（**Custom Configuration**の場合）



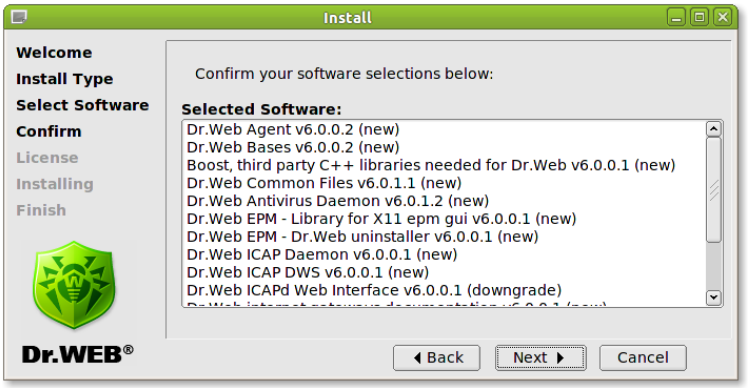
インストールするソフトウェアの選択時に依存関係の確認が自動的に行われます。

例）**DrWeb anti-virus Daemon** を選択すると、**DrWeb Bases** と **DrWeb Common Files** も一緒に選択されます。

"Select All" ボタンを押すと、すべてのコンポーネントが選択され、**"Select None"** ボタンを押すと、すべての選択が解除されます。

3. インストールするコンポーネントの確認を行い、**"Next"** ボタンを押して次に進みます。

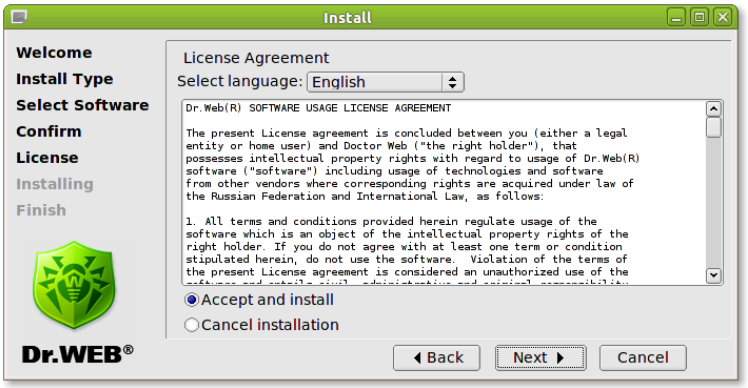
図 4. コンポーネントの確認画面（インストール）



4. ソフトウェア使用許諾契約が表示されます。同意される場合は、**"Accept and install"** を選択し、**"Next"** ボタンを押してインストールを開始します。

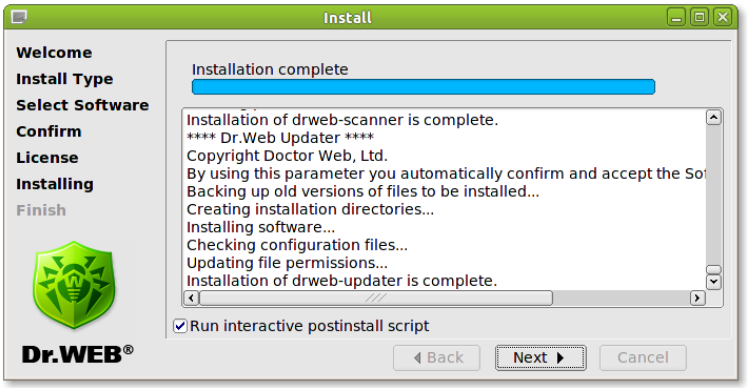
   同意しない場合は、**"Cancel installation"** を選択し、**"Next"** ボタンを押してインストールを終了します。

図 5. ソフトウェア使用許諾契約



5. 上記4でソフトウェア使用許諾契約に同意した場合は、インストール処理が開始し、インストールが完了すると以下の画面が表示されます。

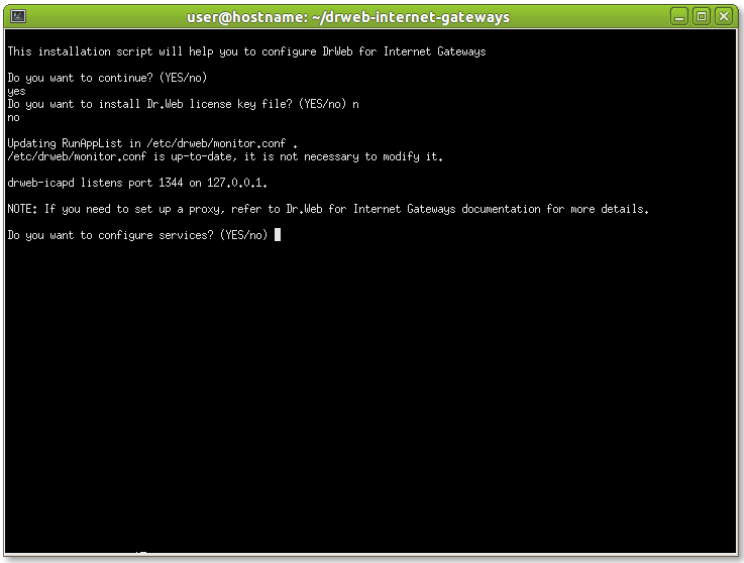
図 6. インストール完了画面



インストール処理のログは、`drweb-internet-gateways_[version number]~[OS name]`ディレクトリのinstall.logファイルに記録されます。

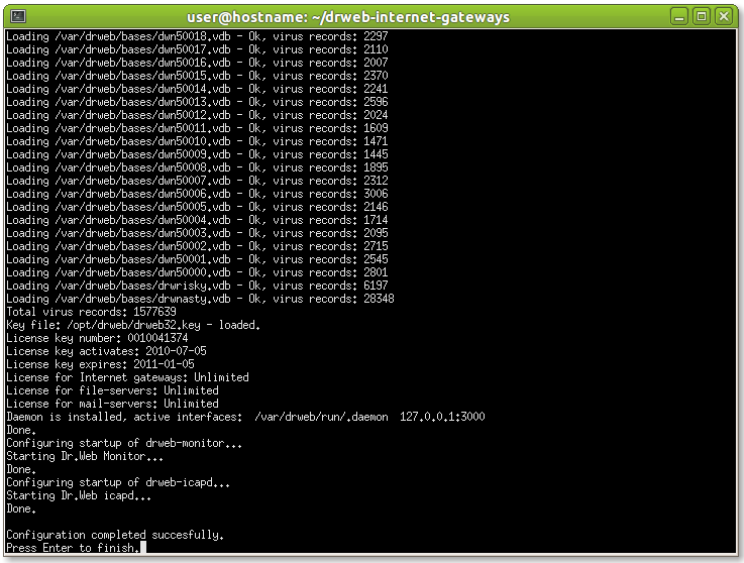
"Run interactive postinstall script" をチェックし、**"Next"** ボタンを押すと、**Dr.Web for Unix Internet gateways**の基本設定を行うスクリプトが起動します。

図 7. インストール後の基本設定（Interactive post-install script）



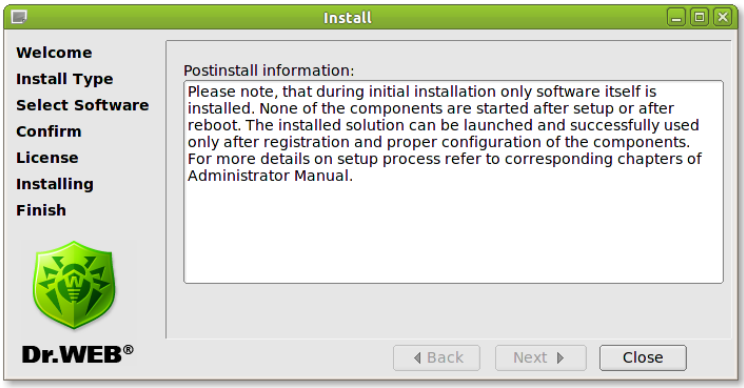
ライセンスキーファイルのパスを指定すると、**Dr.Web for Unix Internet gateways**の動作に必要なサービスが自動的に起動します。

図 8. サービスの開始



6. インストールの終了とインストール後の設定に関する情報が表示されます。

図 9. 終了画面



7. **"Close"** ボタンを押して、GUIインストールを終了します。

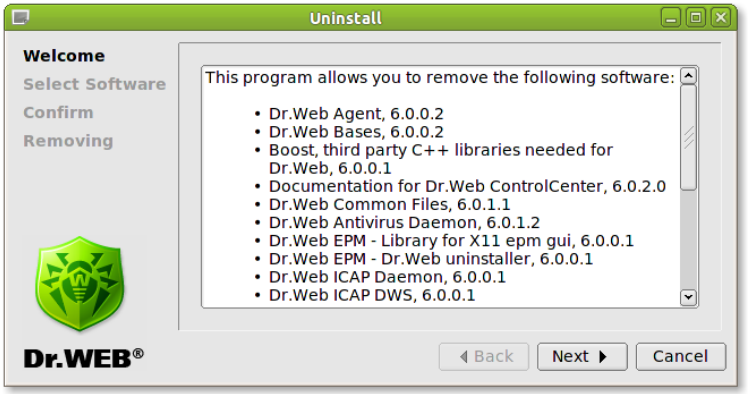
GUIインストーラによるアンインストール

1. 以下のコマンドを実行し、アンインストールを開始します。

```
# drweb-internet-gateways_
[version number]~[OS name]/remove.sh
```

GUIアンインストールの画面が表示されます。

図 10. GUIアンインストール画面

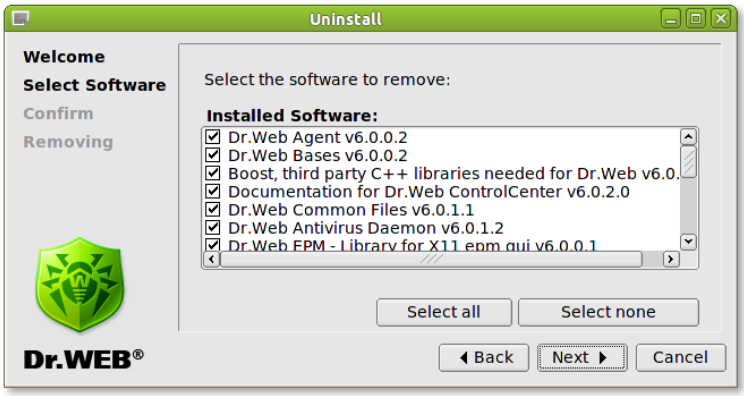


"Next" ボタンを押して次に進みます。アンインストールを終了する場合は、**"Cancel"** ボタンを押します。

2. アンインストールするコンポーネントの選択画面が表示されます。依存関係のあるコンポーネントは自動的に選択されます。

既に他の**Doctor Web**製品がEPM-packagesによって、インストールされていた環境で**Dr.Web for Unix Internet gateways**を使用していた場合、アンインストールするコンポーネントの選択画面に他の**Doctor Web**製品のコンポーネントも表示されます。アンインストールするコンポーネントの選択時に留意してください。

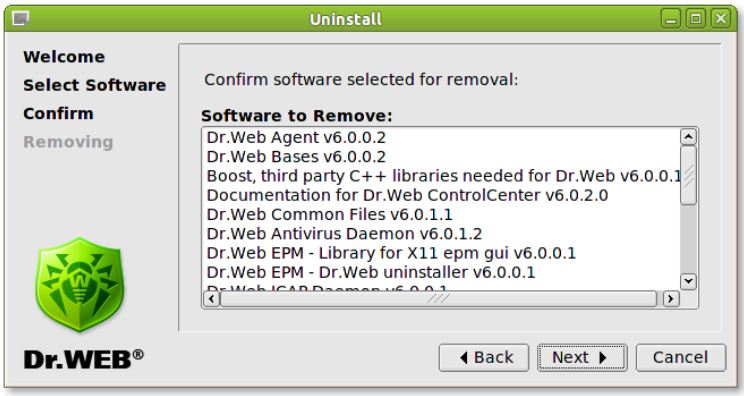
図 11. コンポーネントの選択画面（アンインストール）



"Select All" ボタンを押すと、すべてのコンポーネントが選択され、**"Select None"** ボタンを押すと、すべての選択が解除されます。

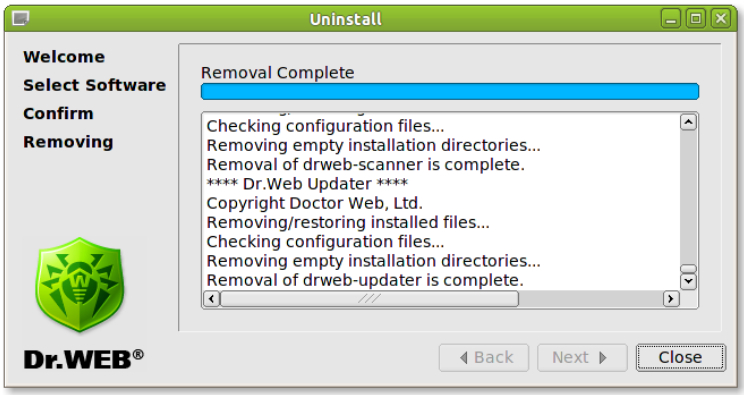
3. アンインストールするコンポーネントの確認を行い、**"Next"** ボタンを押して次に進みます。

図 12. コンポーネントの確認画面（アンインストール）



4. アンインストール処理が開始し、アンインストールが完了すると以下の画面が表示されます。

図 13. アンインストール完了



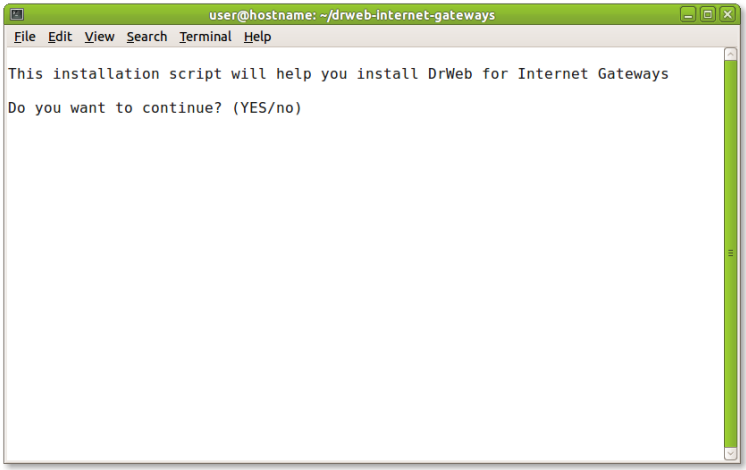
5. **"Close"** ボタンを押して、GUIアンインストールを終了します。

コンソールインストーラによるアンインストール

GUIインストールが行えない場合、自動的にコンソールインストーラが開始されます。また、以下のコマンドで手動でコンソールインストーラを開始することも可能です。

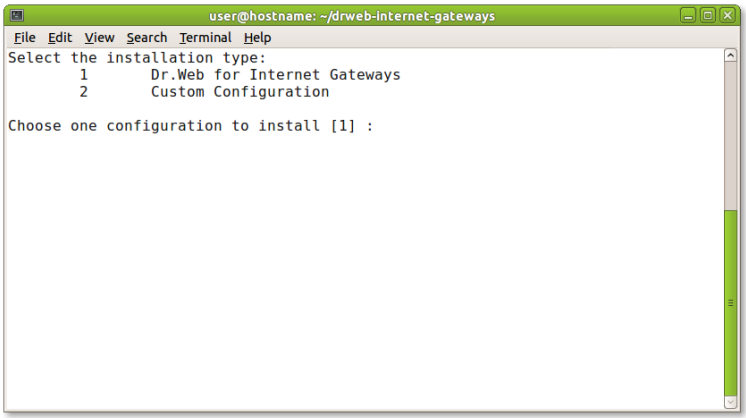
```
# drweb-internet-gateways_[version  number]~[OS name]/setup.sh
```

インストール画面が開きます。

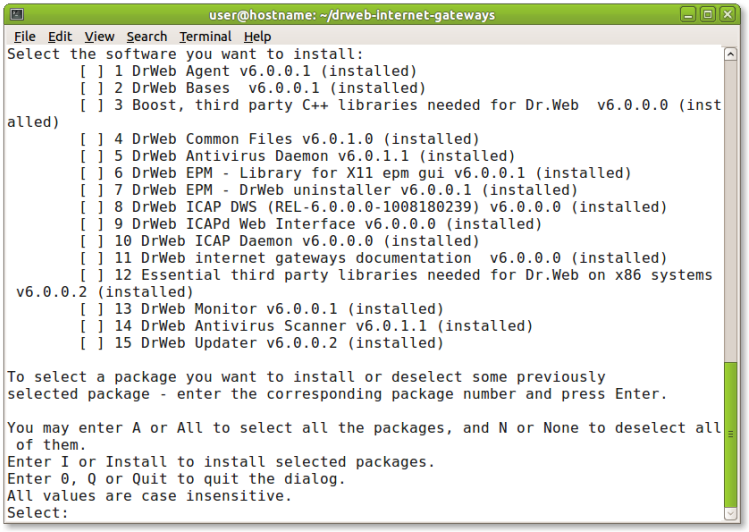


Dr.Web for Unix Internet gatewaysをインストールする場合、**Y** または **Yes** を入力します。インストールしない場合は、**N** または **No** を入力します。(大文字と小文字は区別しません)

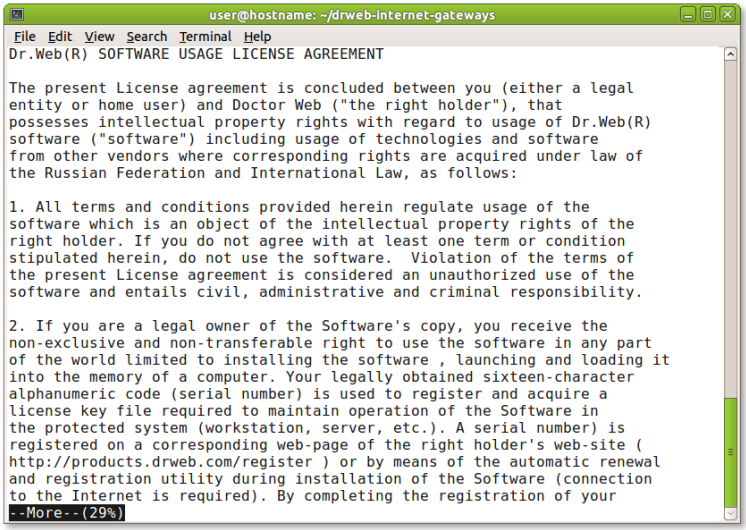
インストールの種別を選択する画面が表示されます。実行したいインストール種別に対応する番号を入力してください。



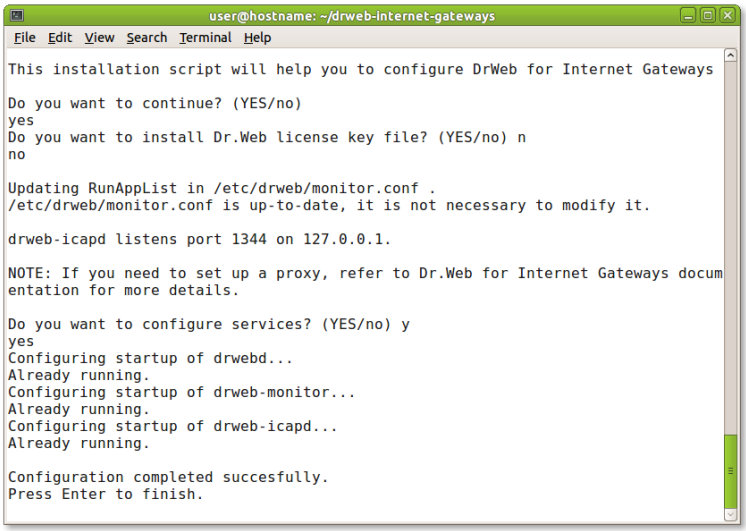
"Custom Configuration" を選択した場合、コンポーネントを選択する画面が表示されます。インストールするコンポーネントに対応する番号を入力してください。



ソフトウェア使用許諾が表示されます。スペースキーでソフトウェア使用許諾のページを進めることができます。



インストールを開始する場合は、ソフトウェア使用許諾に同意し、 **Y** または **Yes** を入力してください。インストールが開始されます。

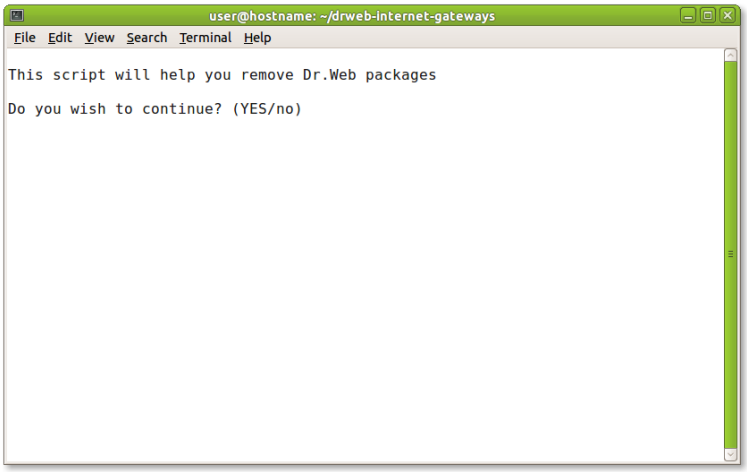


インストール完了後、**Dr.Web for Unix Internet gateways**の基本設定を行うスクリプトが起動します。ライセンスキーファイルのパスを指定し、設定を進めると、**Dr.Web for Unix Internet gateways**の動作に必要なサービスが自動的に起動します。

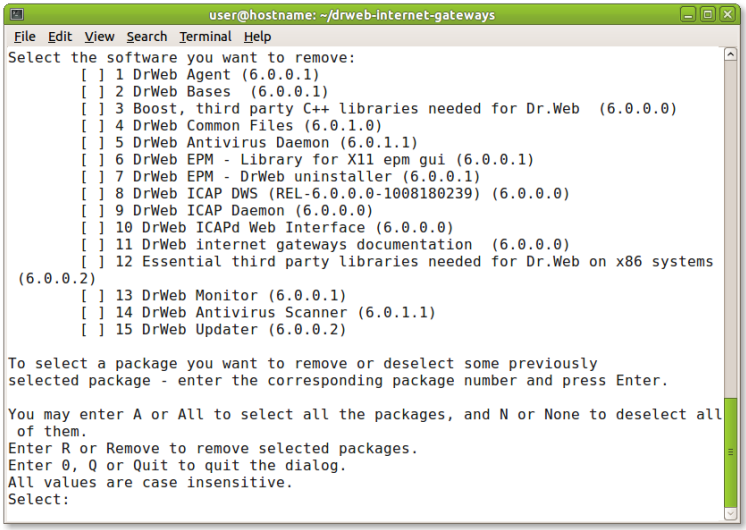
コンソールアンインストーラによるアンインストール

GUIアンインストールが行えない場合、自動的にコンソールアンインストーラが開始します。

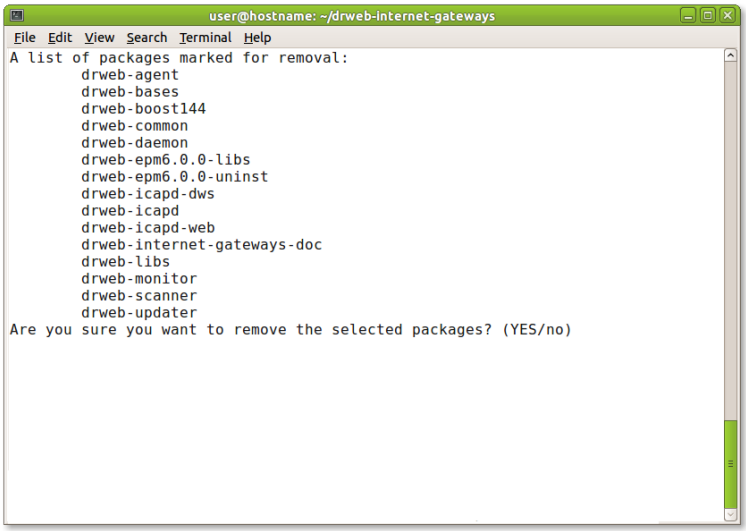
アンインストール画面が開きます。



アンインストールするコンポーネントの選択画面が表示されます。



アンインストールするコンポーネントを選択したら、**Y** または **Yes** ボタンを押してください。（大文字と小文字は区別しません）



アンインストールが開始します。

ネイティブパッケージからのインストール

Dr.Web for UNIX Internet gateways は一般的なLinuxディストリビューション、Solaris、FreeBSDのネイティブパッケージからインストールすることが出来ます。

パッケージは全て**Dr.Web** の公式リポジトリ http://officeshield.drweb.com/drweb/に置かれています。お使いのシステムのパッケージマネージャにこのリポジトリを追加すると、リポジトリからのその他のプログラム同様、必要なパッケージをインストール・アップデート・アンインストール出来るようになります。依存関係は自動的に解決されます。

アップデートを反映させるには、リポジトリからアップデートした後に全てのDr.Webサービスを再起動する必要があります。

Debian、Ubuntu (apt)

Debianリポジトリはライセンスキーによってデジタル署名されています。正常に動作する為に、キーを以下のコマンドでインポートする必要があります。

```
wget -O - http://officeshield.drweb.com/drweb/
drweb.key | apt-key add -
```

または

```
curl http://officeshield.drweb.com/drweb/drweb.
key | apt-key add -
```

お使いのシステムにリポジトリを追加するには `/etc/apt/sources.list` ファイルに以下のラインを加えてください。

```
deb http://officeshield.drweb.com/drweb/debian stable
non-free
```

Dr.Web for UNIX Internet gateways をインストールするには以下のコマンドを使います。

```
apt-get update

apt-get install drweb-internet-gateways
```

Dr.Web for UNIX Internet gateways をアンインストールするには以下のコマンドを使います。

```
apt-get remove drweb-internet-gateways
```

またはグラフィカルマネージャ（Synapticなど）を使ってパッケージをインストール、アンインストールすることも出来ます。

ALT Linux、PCLinuxOS (apt-rpm)

お使いのシステムにリポジトリを追加するには `/etc/apt/sources.list` ファイルに以下のラインを加えてください。

32-bit版:

```
rpm http://officeshield.drweb.com/drweb/
altlinux stable/i386 drweb
```

64-bit版:

```
rpm http://officeshield.drweb.com/drweb/
altlinux stable/x86_64 drweb
```

Dr.Web for UNIX Internet gateways をインストールするには以下のコマンドを使います。

```
apt-get update

apt-get install drweb-internet-gateways
```

Dr.Web for UNIX Internet gateways をアンインストールするには以下のコマンドを使います。

```
apt-get remove drweb-internet-gateways
```

またはグラフィカルマネージャ（Synapticなど）を使ってパッケージをインストール、アンインストールすることも出来ます。

Mandriva (urpmi)

正常に動作する為に、以下のコマンドでキーをインポートする必要があります。

```
rpm --import http://officeshield.drweb.com/
drweb/drweb.key
```

以下のファイルを開きます。

http://officeshield.drweb.com/drweb/drweb-i386.urpmi-media

または

http://officeshield.drweb.com/drweb/drweb-x86_64.urpmi-media

リポジトリをシステムに加えるよう促されます。

またはコンソールを使用して以下のコマンドでリポジトリを追加することも出来ます。

```
urpmi.addmedia drweb http://officeshield.drweb.
com/drweb/mandriva/stable/i386/
```

または

```
urpmi.addmedia drweb http://officeshield.drweb.
com/drweb/mandriva/stable/x86_64/
```

Dr.Web for UNIX Internet gateways をインストールするには以下のコマンドを使います。

```
urpmi.update drweb

urpmi drweb-internet-gateways
```

Dr.Web for UNIX Internet gateways をアンインストールするには以下のコマンドを使います。

```
urpme drweb-internet-gateways
```

またはグラフィカルマネージャ（rpmdrakeなど）を使ってパッケージをインストール、アンインストールすることも出来ます。

Red Hat Enterprise Linux、Fedora、CentOS (yum)

以下のコンテンツのファイルを /etc/yum.repos.d ディレクトリに加えてください。

32-bit版:

```
[drweb]
name=DrWeb - stable
baseurl=http://officeshield.drweb.com/drweb/el5/stable/i386/
gpgcheck=1
enable=1
gpgkey=http://officeshield.drweb.com/drweb/drweb.key
```

64-bit版:

```
[drweb]
name=DrWeb - stable
baseurl=http://officeshield.drweb.com/drweb/el5/stable/x86_64/
gpgcheck=1
enable=1
gpgkey=http://officeshield.drweb.com/drweb/drweb.key
```

Dr.Web for UNIX Internet gateways をインストールするには以下のコマンドを使います。

```
yum install drweb-internet-gateways
```

Dr.Web for UNIX Internet gateways をアンインストールするには以下のコマンドを使います。

```
yum remove drweb-internet-gateways
```

またはグラフィカルマネージャ（PackageKit、Yumexなど）を使ってパッケージをイン

ストール、アンインストールすることも出来ます。

Zypper package manager (SUSE Linux)

リポジトリを追加するには以下のコマンドを実行してください。

```
zypper ar -t YUM http://officeshield.drweb.com/drweb/el5/stable/i386/ drweb
```

または

```
zypper ar -t YUM http://officeshield.drweb.com/drweb/el5/stable/x86_64/ drweb
```

Dr.Web for UNIX Internet gateways をインストールするには以下のコマンドを使います。

```
zypper refresh

zypper install drweb-internet-gateways
```

Dr.Web for UNIX Internet gateways をアンインストールするには以下のコマンドを使います。

```
zypper remove drweb-internet-gateways
```

またはグラフィカルマネージャ(YaSTなど)を使ってパッケージをインストール、アンインストールすることも出来ます。

FreeBSD

FreeBSDのメタポートから**Dr.Web**製品をインストールすることが出来ます。http://officeshield.drweb.com/drweb/freebsd/ports/から必要なポートのアーカイブをダウンロードしてください。アーカイブを解凍し、`make install`を実行します。

Solaris

Solaris のネイティブパッケージはパブリックFTPサーバftp://ftp.drweb.com/pub/drweb/unix/release/Solaris/packages からダウンロードし、`pkgadd`ユーティリティーを使用してインストールすることが出来ます。

Dr.Web for UNIX Internet Gatewaysの起動

Linux and Solarisの場合

Dr.Web for Unix Internet gateways を起動するには以下の手順を実行します。

1. %bin_dir(デフォルト)にdrweb32.key ファイルを配置します。
2. %etc_dirのdrwebd.enableファイルに"1"を指定し、**Dr.Web Daemon**を有効にします。(デフォルト値は、"0"無効です。)
3. %etc_dirのdrweb-monitor.enableファイルに"1"を指定し、**Dr.Web Monitor**を有効にします。
4. initスクリプトを実行し、**Dr.Web Daemon**と**Dr.Web Monitor**を起動します。

```
# /etc/rc.d/init.d/drwebd start
# /etc/rc.d/init.d/drweb-monitor start
```

5. プロキシサーバを起動します。

FreeBSDの場合

Dr.Web for Unix Internet gateways を起動するには以下の手順を実行します。

1. %bin_dir(デフォルト)にdrweb32.key ファイルを配置します。
2. 以下の記述を/etc/rc.confファイルに追加します。
   - drwebd_enable="YES" - **Dr.Web Daemon**を有効にします。

- drweb_monitor_enable="YES" – Dr.Web Monitorを有効にします。

3. initスクリプトを実行し、Dr.Web DaemonとDr.Web Monitorを起動します。
4. プロキシサーバを起動します。

SELinux OS

SELinuxが有効となっているOS上でDr.Web ScannerとDr.Web Daemonを稼動させる場合、drweb-scannerとdrweb-daemonモジュールの動作を可能とするために以下のような手順を実行する必要があります。

設定は、 policygentoolコマンドを使用することができます。(SELinxの設定に関する詳細は、各種Linuxディストリビューションのドキュメントを参照してください。)

例:

```
# policygentool drweb-scanner %bin_dir/drweb.real - Dr.Web Scanner

# policygentool drweb-daemon %bin_dir/drwebd.real - Dr.Web Daemon
```

次の3つのファイルが作成されます。

```
[module_name].te
[module_name].fc
[module_name].if
```

[module_name].te ファイルをコンパイルするために以下のコマンドを実行します。

```
checkmodule -M -m -o module-name [module_name].te
```

ポリシーのコンパイルを行うためには、ご利用のシステムにcheckpolicy パッケージがインストールされている必要があります。

必要なポリシーのコンパイルを行うために以下のコマンドを実行します。

```
semodule_package -o [module_name].pp -m module-
name
```

コンパイルしたポリシーモジュールをインストールするために以下のコマンドを実行します。

```
semodule -i [module_name].pp
```

ソフトウェア登録およびライセンスキーファイル

Dr.Web for Unix Internet gatewaysの使用に関する権利は、ライセンスキーファイルによって制御されています。

ライセンスキーファイルには以下の情報が含まれています。

- 使用を許可された**Dr.Web for Unix Internet gateways**のコンポーネント一覧
- ライセンスの有効期限
- その他の制限事項（保護するワークステーション数など）

ライセンスキーファイルは、"`.key`" の拡張子を持っており、**Dr.Web for Unix Internet gateways**の%bin_dir（デフォルト）に配置します。

ライセンスキーファイルは、不正な改変を防ぐためにデジタル署名されており、変更されたライセンスキーファイルは無効になります。ライセンスキーファイルをテキストエディタで開くと破損する恐れがありますので注意してください。

Dr.Web for Unix Internet gatewaysを購入したパートナー、または**Doctor Web**からライセンスキーファイルを入手します。ライセンスキーファイルには、企業名（またはユーザ名）および販社の名前を含んでいます。

ライセンスキーファイルは、"`.key`" の拡張子または、ライセンスキーを含むzipファイルで提供されます。

ライセンスキーファイルは、次のいずれかの方法で受け取ります。

- ライセンスキーファイルを含むzipファイルを電子メールで受信する。

  （**Doctor Web**のWebサイトで登録した場合）
- インストールパッケージに含まれている場合
- 個別にライセンスキーファイル（"`.key`" 拡張子のファイル）が提供される場合

通常、Webサイトによるシリアル登録後に電子メールでライセンスキーファイルが送られます。入力フォームに従い、シリアル番号を登録してください。ライセンスがアクティベートされ、シリアル番号に対応したライセンスキーファイルが発行されます。

http://products.drweb.co.jp/register/

Dr.Web for Unix Internet gatewaysの再インストールか修復を行う場合、既存のライセンスキーを使用することが推奨されます。ライセンスキーファイルが破損、または紛失している場合は、ライセンスのアクティベートと同じ手順でリカバリすることができます。この場合、シリアル番号などの登録情報は、最初に実施したときと同じ内容にする必要があります。ただし、電子メールアドレスだけは変更可能です。(電子メールアドレスを変更した場合、ライセンスキーファイルは新たに登録した電子メールアドレス宛に送付されます。)登録内容が正しければ、シリアル番号に対応したライセンスキーファイルが発行されます。

同じシリアル番号による登録は、25回まで行うことができます。25回を超えて、ライセンスキーファイルのリカバリを行う場合は、http://support.drweb.com/request/ でライセンスキーファイルのリカバリを要求する必要があります。この場合も必要情報はすべて登録する必要があります。承認された場合、ライセンスキーファイルが電子メールで送付されます。

ライセンスキーファイルの配置場所は、設定ファイル(`drweb32.ini`)の**Key**パラメータで指定されています。

例:

```
Key = %bin_dir/drweb32.key
```

ライセンスキーファイルの読み込みに失敗した場合(パスの誤り、アクセス権の問題など)や有効期限切れの場合、各コンポーネントは終了します。

ライセンスの有効期限まで2週間になると、**Dr.Web Scanner**は警告メッセージを出力します。

また、**Dr.Web Daemon**は、**Daemon**の開始、再起動、再読み込みの度に電子メールで通知します。(`drweb32.ini`の`[Daemon]`セクション**MailCommand**パラメータ で指定されている場合)

Dr.Web Updater

Dr.Web Updaterは、**Dr.Web for Unix Internet gateways**のウイルス定義ファイルとURLフィルタリングのブラックリストとホワイトリストを自動更新するためのPerlスクリプトです。(update.pl)

update.plは、**Dr.Web for Unix Internet gateways**の%bin_dirにあります。

Dr.Web Updaterの設定は、`%etc_dir/drweb32.ini`ファイルの`[Updater]`セクションで定義されています。

update.plの実行方法:

```
$ %bin_dir/update.pl [parameters]
```

データベース更新

Dr.Web for Unix Internet gatewaysを最適な状態で使用するには、ウイルス定義ファイルとURLフィルタリングのブラックリスト、ホワイトリストを定期的に更新する必要があります。

ウイルス定義ファイルは、"`.vdb`"の拡張子です。

ウイルス定義ファイルの更新には、デイリーアップデート(`drwtoday.vdb`)とウィークリーアップデート(`drwXXXYY.vdb`)があります。`XXX`は、ウイルス検査エンジンのバージョンを示しており、YYは00から始まる連番です。(例: `drw60000.vdb`)

デイリーアップデートでは、日々発見される新種のウイルスに対応するために、随時リリースされています。(一日一回以上、頻繁にリリースされています。)新しいデイリーアップデートが適用された場合、`drwtoday.vdb`ファイルは上書きされます。また、新しいウィークリーアップデートが適用された場合、`drwtoday.vdb`ファイルの内容は`drwXXXYY.vdb`ファイルにコピーされ、`drwtoday.vdb`ファイルは空ファイルとして新たに生成されます。

ウイルス定義ファイルは、**Dr.Web for Unix Internet gateways**の%

var_dir/bases/（デフォルト）に配置されます。

drwrisky.vdb, drwnasty.vdbの2つ追加ファイルで、アドウェアやダイヤラー、ハッキングプログラムなどの不正プログラムに対応する定義ファイルを提供します。拡張子や形式は、ウイルス定義ファイルと同様です。

ウイルス、その他の不正プログラムに対応する定義ファイルには、以下のような種類があります。

- drwebase.vdb – 製品リリース時に同梱されるウイルス定義ファイル
- drwXXXYY.vdb – ウイルス定義ファイルのウィークリーアップデート（drwtoday.vdbの一週間分を集約したファイル）
- drwtoday.vdb – ウイルス定義ファイルのデイリーアップデート
- drwnasty.vdb – 製品リリース時に同梱されるマルウェア定義ファイル
- dwnXXXYY.vdb – マルウェア定義ファイルのウィークリーアップデート（dwntoday.vdbの一週間分を集約したファイル）
- dwntoday.vdb – マルウェア定義ファイルのデイリーアップデート
- drwrisky.vdb – 製品リリース時に同梱されるリスクウェア定義ファイル
- dwrXXXYY.vdb – リスクウェア定義ファイルのウィークリーアップデート（dwrtoday.vdbの一週間分を集約したファイル）
- dwrtoday.vdb – リスクウェア定義ファイルのデイリーアップデート

URLフィルタリングのブラックリストとホワイトリストの拡張子は、".dws"です。

- dwfXXXNN.dws – ブラックリストです。XXXには、prn（porno）やmlw（malware）などの3文字が入り、NNにはインデックス番号が入ります。
- white_dwfXXX.dws – ホワイトリストです。ブラックリストと同様にXXXには、prn（porno）やmlw（malware）などの3文字が入ります。

ウイルス定義ファイルの配信サーバは、これらのファイルをlzmaアーカイブで保持しています。

Cronの設定

Linuxの場合: インストール時に `/etc/cron.d/` ディレクトリにウイルス定義ファイルを更新するためのタスクが登録されます。

FreeBSD および Solarisの場合: cronタスクを手動で登録してください。

FreeBSDの場合は、drwebユーザのcrontabに以下のようなタスクを登録してください。

```
*/30 * * * * /usr/local/drweb/update.pl
```

Solarisの場合は、以下のようなコマンドでタスクを登録してください。

```
# crontab -e drweb
# 0,30 * * * * /opt/drweb/update.pl
```

コマンドラインパラメータ

Dr.Web Updaterのパラメータは、設定ファイルに定義する方法とコマンドラインパラメータで指定する方法があります。

- `--ini=path_to_configuration_file`
- `--what=component_to_be_updated`

`Component_to_be_updated`は、`scanner`または、`daemon`です。この値が指定されない場合、**Updater**は設定ファイルの情報を使用します。

また、コマンドラインパラメータとして `--not-need-reload`パラメータを指定することができます。`--not-need-reload`パラメータが指定されていない場合、update.plの処理が終了するとすべてのデーモンが再起動されます。

`--not-need-reload` パラメータには、デーモン名を指定することができます。カンマ区切りで複数指定が可能で、大文字と小文字は区別しません。指定されたデーモンは、再起動されません。

例:

```
$ %bin_dir/update.pl --not-need-reload=drwebd
```

設定ファイル

Updaterの設定は、要件や状況に応じて設定を変更することができます。
Updaterの設定は、設定ファイルに保存されています。(デフォルト: `%etc_dir/drweb32.ini`)

`[Updater] section`

UpdatePluginsOnly = {Yes | No}	Yesが指定された場合、プラグインのみが更新されます。**Daemon**および**Scanner** は更新されません。 デフォルト値: **UpdatePluginsOnly** = No
Section = {Daemon | Scanner}	**Daemon**と**Scanner** のどちらを更新対象とするかの指定です。 コマンドラインパラメータの `--what` で指定された場合、`--what`オプションが優先されます。 デフォルト値: **Section** = Scanner
ProgramPath = {path to file}	**Daemon**または、**Scanner** のパスです。**Dr.Web Updater**が製品バージョンやAPI情報を取得するために使用します。 デフォルト値: **ProgramPath** = %bin_dir/drwebd
SignedReader = {path to file}	電子署名されたファイルの検証用プログラムのパスの指定です。 デフォルト値: **SignedReader** = %bin_dir/read_signed

LzmaDecoderPath = {path to file}	lzmaアーカイブを展開するプログラムのパスの指定です。 デフォルト値: **LzmaDecoderPath** = %bin_dir
LockFile = {path to file}	Dr.Web Updater実行時のロック用ファイルのパスの指定です。 デフォルト値: **LockFile** = %var_dir/run/update.lock
CronSummary = {Yes | No}	Yesの場合、Dr.Web Updaterの実行結果は標準出力に出力されます。Updaterがcronによって実行されている場合、管理者にメールで通知するためにこのモードを使用することができます。 デフォルト値: **CronSummary** = Yes
DrlFile = {path to file}	ウイルス定義ファイルやウイルス検査エンジンなどを更新するための更新サーバのURLが定義されたリストの指定です。Dr.Web Updaterは、リストからランダムでサーバを選択し、更新を行います。このリストは、Dr.Webによって電子署名されているため編集しないでください。 デフォルト値: **DrlFile** = %var_dir/bases/update.drl
CustomDrlFile = {path to file}	カスタムdrlファイル(*.drl)のパスの指定です。Dr.Webによって電子署名されているため編集しないでください。 デフォルト値: **CustomDrlFile** = %var_dir/bases/custom.drl

FallbackToDrl = {Yes | No}	どの*.drl ファイルを最初に使用するかの指定です。Yesの場合、**Updater**は、**CustomDrlFile**で指定されたリストを使用します。失敗した場合、**DrlFile**で指定されたリストが使用されます。 デフォルト値: **FallbackToDrl** = Yes
DrlDir = {path to directory}	プラグインを更新するための更新サーバのURLが定義されたリストを含むディレクトリの指定です。リストは、**Dr.Web** によって電子署名されているため編集しないでください。 デフォルト値: **DrlDir** = %var_dir/drl/
Timeout = {numerical value in seconds}	更新時のダウンロードにおけるタイムアウト(秒)の指定です。 デフォルト値: **Timeout** = 90
Tries = {numerical value}	**Dr.Web Updater**が更新サーバに接続を試みる回数の指定です。 デフォルト値: **Tries** = 3
ProxyServer = {proxy server name or IP}	プロキシサーバの指定です。プロキシサーバを使用して更新する場合は、プロキシサーバのホスト名または、IPアドレスを指定してください。 デフォルト値: **ProxyServer** =
ProxyLogin = {proxy server user login}	プロキシサーバの認証に用いるユーザ名の指定です。 デフォルト値: **ProxyLogin** =

ProxyPassword = {proxy server user password}	プロキシサーバの認証に用いるユーザのパスワードの指定です。 デフォルト値: **ProxyPassword** =
LogFileName = {path to file or syslog}	ログファイルの指定です。syslogを指定することができます。 デフォルト値: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	syslogのファシリティの指定です。 デフォルト値: **SyslogFacility** = Daemon
LogLevel = {Debug | Verbose | Info | Warning | Error | Quiet}	ログの詳細レベルの指定です。 デフォルト値: **LogLevel** = Info
LotusdPidFile = {path to file}	**Lotus Daemon** PID ファイルのパスの指定です。 デフォルト値: **LotusdPidFile** = %var_dir/run/drweblotusd.pid
MaildPidFile = {path to file}	drweb-maild PID ファイルのパスの指定です。 デフォルト値: **MaildPidFile** = %var_dir/run/drweb-maild.pid
IcapdPidFile = {path to file}	drweb-icapd PID ファイルのパスの指定です。 デフォルト値:

	IcapdPidFile = %var_dir/run/drweb_icapd.pid
BlacklistPath = {path to directory}	.dwsフィルが保存されるディレクトリの指定です。 デフォルト値: **BlacklistPath** = %var_dir/dws
AgentConfPath = {path to file}	**Agent**の設定ファイルのパスです。 デフォルト値: **AgentConfPath** = %etc_dir/agent.conf
PathToVadeRetro = {path to file}	libvaderetro.so ライブラリのパスです。 デフォルト値: **PathToVadeRetro** = %var_dir/lib/libvaderetro.so
ExpiredTimeLimit = {number}	ライセンスキーの有効期限について、残り日数を知らせるための指定です。 デフォルト値: **ExpiredTimeLimit** = 14
ESLockfile = {path to file}	ロックファイルのパスです。ロックファイルが存在している場合、**Dr.Web Updater**はcronによる自動実行はされません。 デフォルト値: **ESLockfile** = %var_dir/run/es_updater.lock

更新プロセス

更新プロセスは、以下の手順で行われます。

1. **Dr.Web Updater**が設定ファイルを読み込みます。

2. **Dr.Web Updater**は、`[Updater]`セクションのパラメータだけでなく、**EnginePath**, **VirusBase**, **UpdatePath**, **PidFile**のパラメータも使用します。
3. **Dr.Web Updater**は、更新サーバに利用可能なアップデートのリストを要求し、対応するlzmaアーカイブをダウンロードします。
   lzmaが見つからない場合、`*.vdb`および`*.dws`形式でダウンロードします。`[Updater]`セクションの**LzmaDecoderPath**パラメータに指定されている解凍ユーティリティによって、lzmaアーカイブからファイルを抽出します。
4. Updating Virus Databasesの章で説明されているように、
   `%var_dir/bases/`（デフォルト）にファイルが配置されます。

Dr.Web Control Agent

Dr.Web Control Agent（**Agent**）は、**Dr.Web for Unix Internet gateways**の様々な設定やステータスを集中管理するために使用するモジュールです。**Dr.Web for Unix Internet gateways**の開始や設定の変更、ステータスの変化などを**Dr.Web Enterprise Security Suite**に通知します。

オペレーションモード

Enterpriseモードでは、**Dr.Web Enterprise Security Suite**を利用して、**Dr.Web for Unix Internet gateways**の集中管理ができます。

Agentは、以下のいずれかのモードで動作します。

- **Standalone** Standaloneモードの場合、ホストの集中管理はありません。設定ファイルやライセンスキーなどはローカルドライブ上に存在し、**Agent**はホスト上で設定・管理されます。
- **Enterprise** Enterpriseモードの場合、 ホストは集中管理サーバの配下で管理されます。**Dr.Web for Unix Internet gateways**の機能や設定は、**Dr.Web ESS**で定義されているセキュリティポリシーに従います。ライセンスキーファイルは**Dr.Web ESS**から受け取るため、ホスト上に配置されているライセンスキーファイルは使用されません。

コマンドラインパラメータ

Agentでは、以下のコマンドラインパラメータを使用することができます。

- `-h, --help` - コマンドラインパラメータのヘルプを表示します。
- `-v, --version` - **Agent** のバージョン情報を表示します。
- `-u, --update-all` - すべてのモジュールを更新します。
- `-f, --update-failed` - 標準モードによる更新に失敗したモジュールを更新します。
- `-C, --check-only` - 設定のチェックのみを行います。
- `-p, --newpwd` - **Dr.Web ESS**にアクセスするためのユーザ名と

パスワードを変更します。

- `-d, --droppwd` - **Dr.Web ESS**にワークステーションを再登録するために、**Dr.Web ESS**に登録したユーザ名とパスワードを破棄します。
- `-c <path to file>, --conf <path to file>` - 設定ファイルへのパスの指定です。(デフォルト以外を指定する場合)
- `-s <path to file>, --socket <socket>` - ソケットの指定です。
- `-P <path to file>, --pid-file <path to file>` - **Agent**のPIDファイルのパスの指定です。
- `-e <path to file>, --export-config <path to file>` - **Dr.Web ESS**に設定をエクスポートします。

設定ファイル

Dr.Web Agentの設定は、`%etc_dir/agent.conf` に定義されています。

[Logging]セクション

`[Logging]`セクションには、**Dr.Web Agent**のロギングに関する設定が定義されています。

`[Logging] section`

`Level` = {`Quiet` | `Error` | `Alert` | `Info` | `Debug`}	Agent のログの詳細レベルの指定です。 デフォルト値: **`Level`** = `Info`
`IPCLevel` = {`Quiet` | `Error` | `Alert` | `Info` | `Debug`}	IPCライブラリのログの詳細レベルの指定です。 デフォルト値: **`IPCLevel`** = `Error`
`SyslogFacility` =	syslogのファシリティの指定です。

{Daemon | Local0 .. Local7 | Kern | User | Mail}	デフォルト値: **SyslogFacility** = Daemon
FileName = {path to file or syslog}	ログファイルの指定です。 syslogを指定することができます。 デフォルト値: **FileName** = syslog

[Agent]セクション

[Agent]セクションには、**Dr.Web Agent**に関する設定が定義されています。

[Agent] section

MetaConfigDir = {path to directory}	drweb-agentのメタ設定ファイル(meta-configuration files)があるディレクトリ名の指定です。**Agent**と**Dr.Web** 製品の他のモジュールが連携するために必要な設定が定義されています。**Dr.Web** によって提供されるため、編集の必要はありません。 デフォルト値: **MetaConfigDir** = %bin_dir/agent/
UseMonitor = {Yes | No}	Yesの場合、**Monitor** が**Dr.Web for Unix Internet gateways**の一部として動作するようにdrweb-agentに通知します。 デフォルト値: **UseMonitor** = Yes
MonitorAddress = {address}	**Agent**と**Monitor**が連携するために使用するソケットの指定です。(**Monitor**の設定ファイルで定義されている**Address**パラメータの値と同じです。)

	デフォルト値: **MonitorAddress** = local:%var_dir/ipc/.monitor
MonitorResponseTime = {time in seconds}	drweb-monitorモジュールからの応答を待つ最大時間(秒)の指定です。指定された時間内にMonitorからの応答がない場合、Agentは、drweb-monitorが起動していないと判断し、Monitorとの接続確立を終了します。 デフォルト値: **MonitorResponseTime** = 5
PidFile = {path to file}	AgentのPIDファイルのパスの指定です。 デフォルト値: **PidFile** = %var_dir/run/drweb-agent.pid

[Server]セクション

[Server]セクションには、Dr.Web for Unix Internet gatewaysのモジュールとDr.Web Agentが連携するための設定が定義されています。

[Server] section

Address = {socket address}	ソフトウェアモジュールとAgentが連携するために使用されるソケットの指定です。カンマ区切りで複数のソケットを指定することができます。 デフォルト値: **Address** = local:%var_dir/ipc/.agent
Threads = {numerical value}	drweb-agentのスレッド数の指定です。Agentにウイルス統計を通知するモジュールの最大同時接続数を制御します。

	デフォルト値: **Threads** = 2
Timeout = {time in seconds}	**Agent**と他の**Dr.Web**モジュールが接続を確立する際のタイムアウト値 デフォルト値: **Timeout** = 15

[EnterpriseMode]セクション

[EnterpriseMode]セクションには、**Agent**が**Enterprise**モードで動作するための設定が定義されています。

[EnterpriseMode] section

UseEnterpriseMode = {Yes | No}	Yesの場合、drweb-agentは"Enterprise mode"で動作し、Noの場合は"Standalone mode"で動作します。 デフォルト値: **UseEnterpriseMode** = No
ComputerName = {text value}	**Dr.Web ESS**ネットワークでのコンピュータ名の指定です。 デフォルト値: **ComputerName** =
VirusbaseDir = {path to directory}	ウイルス定義ファイルが配置されているディレクトリの指定です。 デフォルト値: **VirusbaseDir** = %var_dir/bases
PublicKeyFile = {path to file}	**Dr.Web ESS**サーバへの接続に使用する公開鍵のパスの指定です。 デフォルト値:

	PublicKeyFile = %bin_dir/drwcsd.pub
ServerHost = {IP address}	Dr.Web ESSサーバのIPアドレスの指定です。 デフォルト値: **ServerHost** = 127.0.0.1
ServerPort = {port number}	Dr.Web ESSサーバのポート番号の指定です。 デフォルト値: **ServerPort** = 2193
CryptTraffic = {Yes | Possible | No}	Dr.Web ESSサーバとAgent間のトラフィックの暗号化に関する指定です。 デフォルト値: **CryptTraffic** = possible
CompressTraffic = {Yes | Possible | No}	Dr.Web ESSサーバとAgent間のトラフィックの圧縮に関する指定です。 デフォルト値: **CompressTraffic** = possible
CacheDir = {path to directory}	設定ファイルやDr.Web ESSサーバへの登録情報などが保存されるディレクトリの指定です。 デフォルト値: **CacheDir** = %var_dir/agent

[StandaloneMode]セクション

[StandaloneMode]セクションには、Agentが**Standalone**モードで動作するための設定が定義されています。

[StandaloneMode] section

StatisticsServer = {server address}	ウイルス統計サーバのIPアドレスまたは、ホスト名の指定です。 デフォルト値: **StatisticsServer** = stat.drweb.com:80/update
StatisticsUpdatePeriod = {time in minutes}	統計情報の更新レートの指定です。 デフォルト値: **StatisticsUpdatePeriod** = 10
StatisticsProxy = {proxy server address}	ウイルス統計プロキシサーバのIPアドレスまたは、ホスト名の指定です。 **例:** StatisticsProxy = localhost:3128 デフォルト値: **StatisticsProxy** =
StatisticsProxyAuth = {text value}	プロキシサーバへのアクセスに利用するユーザ名とパスワードの指定です。 **例:** StatisticsProxyAuth = test:testpwd デフォルト値: **StatisticsProxyAuth** =
UUID = {identifier}	ウイルス統計サーバ http://stat.drweb.com/ で利用するUUIDの指定です。UUIDは、統計の転送に必要です。機能を有効にする場合は、ユニークなユーザ識別子としてUUIDを指定する必要があります。 デフォルト値: **UUID** =

LicenseFile = {path to file}	Dr.Webのライセンスキーファイルの指定です。 デフォルト値: **LicenseFile** = %bin_dir/drweb32.key
ProtectedEmails = {lookups}	保護された電子メールアドレスのリストの指定です。 デフォルト値: **ProtectedEmails** = file:%etc_dir/email.ini

[Update]セクション

[Update]セクションには、Dr.Web ESSからDr.Web for Unix Internet gatewaysのコンポーネントを更新するための定義が含まれています。

[Update] section

CacheDir = {path to directory}	Agentがダウンロードした更新ファイルを一時保存するディレクトリの指定です。 デフォルト値: **CacheDir** = %var_dir/agent/cache
RegFile = {path to file}	インストール済みの更新情報に関する指定です。 デフォルト値: **RegFile** = %var_dir/agent.reg
Timeout = {time in seconds}	Agentが更新ファイルをダウンロードする際のタイムアウト値(秒)の指定です。 デフォルト値: **Timeout** = 120

RootDir = {path to directory}	ルートディレクトリの指定です。 デフォルト値: **RootDir** = /
UpdaterFile = {path to directory}	更新ユーティリティのパスの指定です。 デフォルト値: **UpdaterFile = /opt/drweb/update.pl**

詳細については、**Dr.Web ESS**の管理者用ガイドを参照してください。

Dr.Web Unix Control Agentの起動

インストール時の"Configuration Services"の選択でYesを指定している場合は、**Dr.Web Agent**を含むすべてのサービスは自動的に起動しています。

Agentがデフォルト設定で起動すると、以下の処理が実行されます。

1. 設定ファイルを読み込みます。設定ファイルが見つからない場合、**Agent**は終了します。
2. [EnterpriseMode]セクション、[Standalone]セクションのパラメータ設定を参照し、指定されたモードで動作します。
3. **Agent**と**Dr.Web**モジュールが連携するためのソケット作成します。ソケットが作成できない場合、**Agent** は終了します。

設定の反映は、各動作モードに依存します。

集中管理モード(**Enterprise**モード)の場合:

- **Agent**は、**Dr.Web Enterprise Security Suite**に接続します。**Dr.Web Enterprise Security Suite**に接続できない場合や認証プロセスに失敗した場合は、スタンドアロンモードで動作します。
- **Agent**は、**Dr.Web Enterprise Security Suite**から設定情報と鍵ファイルを受信します。すべての設定と鍵ファイルの受信を終えると、**Agent**が動作可能な状態となります。

スタンドアロンモードの場合:

- **Agent**と**Dr.Web**モジュールが連携するためのmeta-configurationファイルがロードされます。meta-configurationファイルの場所は、設定ファイルの`[Agent]`セクションの`MetaConfigDir`パラメータで指定されています。 meta-configurationファイルの読み込みに成功すると、**Agent**が動作可能な状態となります。

他のソフトウェアとの連携

他のソフトウェアとの連携は、**Agent**のmetaconfigurationファイル(amc-files)によって行われます。

`Application`セクションで、それぞれのモジュールに関する設定が定義されています。セクションの終端では、`EndApplication`を指定する必要があります。

以下のパラメータが定義されている必要があります。

- **id**: **Dr.Web ESS**で利用するID
- **ConfFile**: モジュールの設定ファイルへのパス
- **Components**: コンポーネントの定義。終端で、`EndComponents`を指定する必要があります。各コンポーネントの名前やパラメータなど、コンポーネントの動作に必要な設定を指定します。

Dr.Web ICAP for Linux のamc-fileの例:

```
Application "ICAPD"
        id              49
        ConfFile        "/etc/drweb/drweb-
icapd.ini"
        Components
                drweb-icapd             /
_I=Icapd
        EndComponents
EndApplication
```

ウィルス統計情報

Agentは、ウイルス統計情報を**Dr.Web**のウイルス統計サイト http://stat.drweb.com/ に送信します。（インターネット接続が可能な場合）

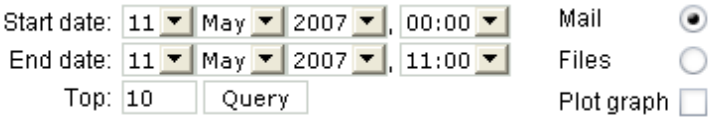
Agentが、Enterpriseモードで動作している場合は、**Dr.Web Dr.Web ESS**に統計情報を送信します。

Agentは、ユニークなユーザ識別子（UUID）で統計サーバに接続する必要があります。デフォルトで、ライセンスキーファイルのMD5チェックサムがUUIDとして使用されます。**Dr.Web**テクニカルサポートサービスに依頼することでUUIDを取得することも可能ですが、この場合は、**Agent**の設定ファイルで明示的にUUIDを指定する必要があります。

http://stat.drweb.com/ のサイトでは、ウイルス統計情報の中から期間中に多く検出されたウイルスを表示させることができます。（検出の総数、割合）

http://stat.drweb.com/ にアクセスすることで、**Dr.Web**によって収集されたウイルス統計情報を参照することができます。

図 14. ウイルス統計情報



11.05.2007 00:00 - 11.05.2007 11:00		
1	Win32.HLLM.Beagle	17570 (29.94%)
2	Win32.HLLM.Netsky.35328	8585 (14.63%)
3	Win32.HLLM.MyDoom.based	5757 (9.81%)
4	Win32.HLLM.Netsky.based	5408 (9.21%)
5	Win32.HLLM.Perf	3873 (6.60%)
6	Win32.HLLM.Graz	3639 (6.20%)
7	Win32.HLLM.MyDoom.33808	3128 (5.33%)
8	Win32.HLLP.Sector	1294 (2.20%)
9	Win32.HLLM.Beagle.pswzip	1092 (1.86%)
10	Win32.HLLM.MyDoom.49	944 (1.61%)

Total scanned: 3638081
Total infected: 58688 (1.61%)

以下のように検索条件を指定して、検索することができます:

1. **Mail** または、 **Files** フラグを指定することで、電子メールメッセージの検出を表示するか、ファイルの検出を表示するかを選択します。
2. **Start date** と **End date** で、検索対象にする期間を指定します。
3. **Top** フィールドを指定します。(TOP:10、TOP:20のように指定します。)
4. グラフィック表示も行いたい場合は、**Plot graph** チェックボックスにチェックを入れます。

5. **Query** ボタンを押します。

ウイルス統計情報は、HTML形式とXML形式が利用できます。
http://info.drweb.com/export/xml/top でウイルス統計情報のXMLフォームを確認することができます。

例:

```
<drwebvirustop        period="24"        top="5"
vdbaseurl="http://info.drweb.com/
virus_description/"       updatedutc="2009-06-09
09:32:02">
    <item>
        <vname>Win32.HLLM.Netsky</vname>
        <dwvlid>62083</dwvlid>
        <place>1</place>
        <percents>34.201062139103</percents>
    </item>
    <item>
        <vname>Win32.HLLM.MyDoom</vname>
        <dwvlid>9353</dwvlid>
        <place>2</place>
        <percents>25.1303270912579</percents>
    </item>
    <item>
        <vname>Win32.HLLM.Beagle</vname>
        <dwvlid>26997</dwvlid>
        <place>3</place>
        <percents>13.4593034783378</percents>
    </item>
    <item>
        <vname>Trojan.Botnetlog.9</vname>
        <dwvlid>438003</dwvlid>
```

```
        <place>4</place>
        <percents>7.86446592583328</percents>
    </item>
    <item>
        <vname>Trojan.DownLoad.36339</vname>
        <dwvlid>435637</dwvlid>
        <place>5</place>
        <percents>7.31494163115527</percents>
    </item>
</drwebvirustop>
```

以下のXML属性が使用されています。

- `period` - ウイルス統計情報の対象期間（時間）
- `top` - ウイルス統計情報で多く検出されたウイルスの順位
- `updatedutc` - ウイルス統計情報の最終更新日時
- `vname` - ウイルス名
- `place` - 統計上のウイルスの場所
- `percents` - 検出の割合

`period` とサンプルの値は、ユーザによって変更することはできません。

パーソナライズされたウイルス統計情報を得る場合は、http://stat.drweb.com/view/<UID> にアクセスします。`<UID>` は、ライセンスキーファイルのMD5チェックサムです。

http://stat.drweb.com/xml/<UID> でパーソナライズされたウイルス統計情報のXMLフォームを確認することができます。`<UID>` は、ライセンスキーファイルのMD5チェックサムです。

例:

```
<drwebvirustop period="24" top="2" user="<UID>"
lastdata="2005-04-12 07:00:00+04">
    <item>
```

```
        <caught>69</caught>
        <percents>24.1258741258741</percents>
        <place>1</place>
        <vname>Win32.HLLM.Netsky.35328</vname>
    </item>
    <item>
        <caught>57</caught>
        <percents>19.9300699300699</percents>
        <place>2</place>
        <vname>Win32.HLLM.MyDoom.54464</vname>
    </item>
</drwebvirustop>
```

以下のXML属性が使用されています。

- `period` - ウイルス統計情報の対象期間（時間）
- `top` - ウイルス統計情報で多く検出されたウイルスの順位
- `user` - ユーザ識別子
- `lastdata` - ユーザが統計サーバに統計情報を送信した最終日時
- `vname` - ウイルス名
- `place` - 統計上のウイルスの場所
- `caught` - 検出数
- `percents` - 検出の割合

`period`とサンプルの値は、ユーザによって変更することはできません。

Dr.Web Monitor

Dr.Web Unix Monitorは、メモリ常駐型のモジュールです。

Dr.Web Unix Monitorは、**Dr.Web for Unix Internet gateways**の耐障害性を向上する役割を担っています。異常発生時のコンポーネントの再起動やソフトウェアモジュールの起動と停止を確実に行います。**Monitor**は、すべてのモジュールを起動し、必要に応じて付加コンポーネントの読み込みを行います。

すべてのモジュールをロードすると、**Monitor**はロードしたモジュールを永続的に制御します。**Monitor**は制御シグナルを送ることで各モジュールと連携し、モジュールまたは、コンポーネントが異常動作した場合は、異常が生じたアプリケーションを再起動します。再起動を試みる回数と時間は、**Monitor**の設定ファイルに定義されています。モジュールの開始に異常が生じた場合、**Monitor**はシステム管理者に通知します。

動作モード

Dr.Web Enterprise Security Suiteを利用して、**Dr.Web for Unix Internet gateways**の集中管理することができます。(**Enterprise**モード)

Monitor は、以下のいずれかのモードで動作します。

- **Standalone** Standaloneモードの場合、ホストの集中管理はありません。設定ファイルやライセンスキーなどはローカルドライブ上に存在し、**Monitor**はホスト上で設定・管理されます。
- **Enterprise** Enterpriseモードの場合、 ホストは集中管理サーバの配下で管理されます。**Dr.Web for Unix Internet gateways**の機能や設定は、**Dr.Web ESS**で定義されているセキュリティポリシーに従います。ライセンスキーファイルは**Dr.Web ESS**から受け取るため、ホスト上に配置されているライセンスキーファイルは使用されません。

Enterpriseモードを使用する場合

1. 集中管理サーバへの接続に必要となる情報(公開鍵や接続先の情報など)をウイルス対策の管理者に確認します。
2. **Monitor**の設定ファイル(デフォルト: `%etc_dir/monitor.conf`)の`UseEnterpriseMode`パラメータに**Yes**を指定してくださ

い。

Dr.Web for Unix Internet gatewaysの集中管理をフルサポートするために、**Agent**の動作もEnterpriseモードにする必要があります。

Standaloneモードを使用する場合

1. **Monitor**の設定ファイル(デフォルト: `%etc_dir/monitor.conf`)に、**Monitor**によって起動するモジュールの一覧を記載します。`[Monitor]`セクションの`RunAppList` パラメータで指定されています。
2. [Monitor]セクションの`UseEnterpriseMode`パラメータに**No**を指定してください。

集中管理サーバから受け取るライセンスキーファイルを使用することはできません。ローカルドライブの所定の場所に**Dr.Web for Unix Internet gateways**の有効なライセンスキーファイルを配置する必要があります。

コマンドラインパラメータ

Dr.Web Monitorでは、以下のコマンドラインパラメータを使用することができます。

- `-h, --help` - コマンドラインパラメータのヘルプを表示します。
- `-v, --version` - **Monitor**のバージョン情報を表示します。
- `-u, --update` - 更新モード
- `-C, --check-only` - 設定のチェックのみを行います。
- `-A, --check-all` - すべてのモジュールの設定をチェックします。
- `-c <path to file>, --conf <path to file>` - 設定ファイルへのパスの指定です。(デフォルト以外を指定する場合)
- `-r, --run component1[,component2]`- コンポーネントを起動します。(指定した順序で起動)

例:

```
-r AGENT
```

設定ファイル

Dr.Web Monitorの設定は、`%etc_dir/monitor.conf` に定義されています。

[Logging]セクション

`[Logging]`セクションには、**Dr.Web Monitor**のロギングに関する設定が定義されています。

`[Logging] section`

Level = {Quiet | Error | Alert | Info | Debug}	**Monitor**のログの詳細レベルの指定です。 デフォルト値: **Level** = Info
IPCLevel = {Quiet | Error | Alert | Info | Debug}	IPCライブラリのログの詳細レベルの指定です。 デフォルト値: **IPCLevel** = Error
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	syslogのファシリティの指定です。 デフォルト値: **SyslogFacility** = Daemon
FileName = {syslog | path to file}	ログファイルの指定です。 syslogを指定することができます。 デフォルト値: **FileName** = syslog

[Monitor]セクション

[Monitor] セクションには、**Monitor**の主要な設定が定義されています。

[Monitor] section

RunForeground = {Yes | No}	Yesの場合、**Monitor**をフォアグラウンドで動作させます。 通常は、デフォルトのNo(デーモン)を使用します。特別なユーティリティ(daemontoolsなど)を使用する場合にYesを指定します。 デフォルト値: **RunForeground** = No
User = {text value}	**Monitor**を実行するユーザ名の指定です。 デフォルト値: **User** = drweb
Group = {text value}	**Monitor**を実行するユーザのグループの指定です。 デフォルト値: **Group** = drweb
PidFileDir = {path to directory}	**Monitor**のPIDファイルが保存されるディレクトリの指定です。 デフォルト値: **PidFileDir** = %var_dir/run/
ChDir = {path to directory}	**Monitor**の作業ディレクトリを変更する場合の指定です。指定した場合、**Monitor** は作業ディレクトリを指定されたディレクトリに変更します。 デフォルト値: **ChDir** = /

MetaConfigDir = {path to directory}	メタ設定ファイル（meta-configuration files）があるディレクトリ名の指定です。**Monitor**と**Dr.Web**製品の他のモジュールが連携するために必要な設定が定義されています。**Dr.Web** によって提供されるため、編集の必要はありません。 デフォルト値: **MetaConfigDir** = %etc_dir/monitor/
Address = {socket address}	**Monitor**が制御シグナルを受信するために使用するソケットの指定です。 デフォルト値: **Address** = local:%var_dir/ipc/.monitor
Timeout = {time in seconds}	**Monitor**と他の**Dr.Web**モジュールが接続を確立する際のタイムアウト値の指定です。 デフォルト値: **Timeout** = 5
TmpFileFmt = {text value}	一時ファイル名の指定です。 例:path_to_file.XXXXXX X - 一時利用する乱数 デフォルト値: **TmpFileFmt** = %var_dir/msgs/tmp/monitor.XXXXXX
RunAppList = {text value}	**Monitor**によって起動するモジュールの指定です。カンマ区切りで複数指定が可能です。 **Dr.Web**モジュールのアンインストール時、このパラメータは変更されません。アンインストール後、手動でアンインストールしたモジュールの指定を削除する必要があります。 デフォルト値:

	RunAppList = AGENT
UseEnterpriseMode = {Yes | No}	Enterpriseモードの指定です。 Yesの場合、Enterpriseモードで動作します。**Monitor**によって起動するモジュールの一覧は、**RunAppList**パラメータからではなく、**Agent** から受け取ります。 Noの場合、Standaloneモードで動作します。 デフォルト値: **UseEnterpriseMode** = No
RecoveryTimeList= {time in seconds}	モジュールを再起動する際のインターバル(秒)の指定です。 カンマ区切りで、1回目のインターバル、2回目のインターバルのように複数の値を指定することができます。 デフォルト値: **RecoveryTimeList** = 0,30,60
InjectCmd = {string}	レポートを送信するコマンドの指定です。 レポートを送信したい場合にアドレスを指定して使用します。 デフォルト値: **InjectCmd** = "/usr/sbin/sendmail -t"
AgentAddress = {socket address}	**Monitor**が**Agent**と連携するために使用する**Agent**のソケットの指定です。(**Dr.Web Agent**の設定ファイルの**Address** パラメータで指定されている値と同じである必要があります。) デフォルト値: **AgentAddress** = local:%var_dir/ipc/.agent

AgentResponseTime = {time in seconds}	drweb-agentモジュールからの応答を待つ最大時間の指定です。 指定時間の間、**Agent**から応答がない場合、**Monitor**はdrweb-agentエージェントが動作していないと判断し、**Agent**の再起動を試みます。
	デフォルト値: **AgentResponseTime** = 5

Dr.Web Unix Monitorの起動

インストール時の"Configuration Services"の選択でYesを指定している場合は、**Dr.Web Monitor**を含むすべてのサービスは自動的に起動しています。

Monitorがデフォルト設定で起動すると、以下の処理が実行されます。

1. **Monitor**は設定ファイルを検索して、読み込みます。設定ファイルが見つからない場合、プロセスは終了します。
2. daemonモードになり、ログファイルに情報を出力します。
3. 他のソフトウェアモジュールと**Monitor**が連携するためのソケットを作成します。ソケットが作成できない場合は、プロセスは終了します。
4. PIDファイルを作成します。PIDファイルを作成できない場合は、終了します。
5. drweb-monitorモジュールが他のソフトウェアモジュールを起動します。モジュールの起動に失敗した場合、**Monitor**は再起動を試みます。

自動モードによるDr.Web Monitorの起動が成功している場合:

- %etc_dir/drweb-monitor.enable ファイルに"1"が指定されています。(Linux と Solarisの場合)
- /etc/rc.conf ファイルにdrweb_monitor_enable="YES"の記述が追加されています。(FreeBSDの場合)

他のソフトウェアとの連携

他のソフトウェアとの連携は、mmc-filesによって行われます。mmc-filesは、**Dr.Web Monitor**と連携できる製品パッケージに含まれています。

Applicationセクションで、それぞれのモジュールに関する設定が定義されています。セクションの終端では、EndApplicationを指定する必要があります。

以下のパラメータが定義されている必要があります。

- **FullName** - コンポーネントのフルネーム
- **Path** - バイナリファイルへのパス
- **Depends** - 先に起動している必要があるコンポーネントの名前（例. DAEMONを起動する場合には、AGENTが起動している必要があります。）
  依存関係がない場合は、このパラメタをスキップすることができます。
- **Components** - コンポーネントの定義。終端で、EndComponents を指定する必要があります。各コンポーネントの名前やパラメータなど、コンポーネントの動作に必要な設定を指定します。

Dr.Web Daemon for Linuxのmmc-fileの例:

```
Application "DAEMON"
    FullName    "Dr.Web (R) Daemon"
    Path        "/opt/drweb/"
    Depends     "AGENT"
    Components
        # name   args   maxStartTime
maxStopTime     NotifyType  UserGroup
        drwebd  "-a=local:/var/drweb/ipc/.agent
--foreground=yes"  30 10 MAIL drweb:drweb
    EndComponents
EndApplication
```

コマンドラインDr.Web Scanner

Dr.Web Scannerは、ローカルマシン上のマルウェアを検出するコマンドラインスキャナです。

コマンドラインパラメータ

Dr.Web Scannerは、以下のコマンドで実行します。

```
$ %bin_dir/drweb <path> [parameters]
```

<path> にはウイルス検査を実行するディレクトリを指定します。パラメータを指定しないで**Scanner**を実行した場合、デフォルトのパラメータ設定で検査を行います。

感染したファイル、または感染が疑われるファイルを検出した場合、検出内容に関する情報を表示します。

```
/path/file infected [virus] VIRUS_NAME
```

また、検査が完了すると、以下のような検査レポートを表示します。

```
Report for "/opt/drweb/tmp":
Scanned         : 34/32      Cured    : 0
Infected        : 5/5        Removed  : 0
Modifications   : 0/0        Renamed  : 0
Suspicious      : 0/0        Moved    : 0
Scanning time   : 00:00:02   Speed    : 5233
KB/s
```

ウイルス検査のテストを行う場合は、製品パッケージに含まれているeicar.rusファイルを使用することができます。eicar.rusファイルをテキストエディタで開き、記載内容に従ってeicar.comファイルに変更してください。

Dr.Web Scannerでeicar.comファイルを検査すると以下のようなメッセージが

出力されます。

```
%bin_dir/doc/eicar.com infected by Eicar Test File (Not a Virus!)
```

Eicar Test Fileはウイルスではなく、アンチウイルス製品のテストに使用されている無害な68バイトのコードです。

Scannerには多くのコマンドラインパラメータがあります。-arのようにハイフン("-")とパラメータを組み合わせて使用し、空白(スペース)で区切ることでパラメータを指定します。

Scannerで利用可能なオプションは、-?, -h, -helpパラメータを指定することで確認できます。

```
$ %bin_dir/drweb -h
```

コマンドラインパラメータの主な内容

- 検査対象の指定
- 検査内容の指定
- 検出時の動作の指定
- スキャナ、検査結果の出力に関する指定

検査対象の指定:

- `path` – ウイルス検査を実行するパスを指定します。複数のパスを指定することが可能です。
- `@[+]<file>` – ファイルに記載されたオブジェクトを検査します。
- `sd` – 検査対象ディレクトリのサブディレクトリ、ファイルを検査します。
- `fl` – シンボリックリンク先のファイル・ディレクトリを検査します。

検査内容の指定:

- `al` – すべてのファイルを検査します。
- `ar[d|m|r][n]` – アーカイブファイルを検査します。(ARJ, CAB, GZIP, RAR, TAR, ZIP etc)
- `cn[d|m|r][n]` – check files in containers(HTML, RTF, PowerPoint etc)
- `ml[d|m|r][n]` – 電子メール書式ファイルを検査します。

- upn – 圧縮された実行ファイルを検査します。LZEXE, DIET, PKLITE, EXEPACK
- ex – 設定ファイルの**FilesTypes**パラメータで指定されている拡張子のファイルを検査します。
- ha – ヒューリスティック解析を有効にします。

d 削除
m 隔離
r 名前変更
n アーカイブなどの形式に関する情報を出力しません

検出時の動作の指定:

- cu[d|m|r] – 感染ファイルの修復
- ic[d|m|r] – 修復不可能なファイルに対する動作
- sp[d|m|r] – 感染が疑われるファイルに対する動作
- adw[d|m|r|i] – アドウェアに対する動作
- dls[d|m|r|i] – ダイヤラーに対する動作
- jok[d|m|r|i] – ジョークプログラムに対する動作
- rsk[d|m|r|i] – リスクウェアに対する動作
- hck[d|m|r|i] – ハッキングプログラムに対する動作

d 削除
m 隔離
r 名前変更
i 無視

スキャナ、検査結果の出力に関する指定:

- v, version – **Scanner**とアンチウイルスエンジンのバージョン情報を表示します。
- ki – ライセンスキーと所有者に関する情報を表示します。(UTF8)
- foreground[yes|no] – **Scanner**をフォアグラウンドで起動するか、バックグラウンドで起動するかを指定します。
- ot – 情報を標準出力に出力します。
- oq – 情報の出力を無効にします。
- ok – 感染していないクリーンなファイルを"Ok" で表示します。

- log=<path to file> – 指定ファイルにログを記録します。
- ini=<path to file> – 設定ファイルへのパスの指定です。
- lng=<path to file> – 言語ファイルへのパスの指定です。
- -a=<Agent address> – Scannerを集中管理モードで開始します。
- --only-key – Agentからライセンスキーファイルのみを受信してScannerを開始します。

以下のパラメータは、パラメータの後ろに"-"を付けることで無効にすることができます。

```
-ar -cu -ha -ic -fl -ml -ok -sd -sp
```

例:

以下のコマンドで検査を実行した場合、ヒューリスティック解析が無効になります。（デフォルト有効）

```
$ drweb <path> -ha-
```

Scanner のデフォルトのパラメータ設定は以下のとおりです。

```
-ar -ha -fl- -ml -sd
```

デフォルトのパラメータ設定は、最適なウイルス検査を実行するための推奨設定となっています。前述の方法により、必要に応じて無効にするパラメータを指定することができますが、アーカイブファイルの検査を無効にした場合にはウイルス対策のレベルが低下する可能性があります。

尚、デフォルトのパラメータ設定では、感染ファイルや感染が疑われるファイルを検出しても修復に関する動作は行いません。感染ファイルを修復するには検出時の動作を明示的に指定する必要があります。

推奨動作は以下のとおりです。

- cu – 修復
- icd – 修復不可能なファイルを削除します。
- spm – 感染が疑われるファイルを隔離します。
- spr – 感染が疑われるファイルを名前変更します。

cu（修復）を指定してScannerを実行した場合、感染ファイルをの修復を試みます。（感染状況やウイルスの種類によっては、修復できないことがあります。）

アーカイブの中の感染ファイルを検出した場合には、修復や削除などの動作は行われません。アーカイブファイルを展開し、動作を指定した上でScannerを実行する必要があります。

d（削除）を指定してScannerを実行した場合、ハードディスク上から感染ファイルを削除します。 このオプションは、修復不可能な感染ファイルに対する動作に適しています。

r（名前変更）を指定してScannerを実行した場合、ファイルの拡張子を変更します。（eicar.#omのように拡張子の最初の文字を"#"に変更します。）感染が疑われるファイルに対する動作に適しており、ファイルが実行されることを防ぐことができます。

m（隔離）を指定してScannerを実行した場合、感染ファイルまたは、感染が疑われるファイルを隔離ディレクトリ（デフォルト: `%var_dir/infected/`）に隔離します。

デイリー検査コマンド（推奨）:

```
$ drweb <path> -cu -icd -spm -ar -ha -fl-
-ml -sd
```

コマンドの内容をテキストファイルに保存し、以下のように実行可能にすることで、検査コマンドをシェルスクリプトで実行させることもできます。

```
# chmod a+x [file name]
```

また、Scannerのデフォルト設定は設定ファイルで変更することができます。

設定ファイル

Scannerの設定は、要件や状況に応じて設定を変更することができます。
Scannerの設定は、設定ファイルに保存されています。（デフォルト: `%etc_dir/drweb32.ini`）

別の設定ファイルを使用する場合は、以下のように -ini パラメータで設定ファイル

をフルパスで指定して、Scannerを実行してください。

```
$ %bin_dir/drweb -ini=%bin_dir/etc/drweb.ini
```

[Scanner] section

EnginePath = {path to file}	ウイルス検査エンジン(drweb32.dll)の指定です。 デフォルト値: **EnginePath** = %var_dir/lib/drweb32.dll
VirusBase = {path to file}	ウイルス定義ファイルの指定です。 ワイルドカード"*"の利用とカンマ","区切りによる複数指定が可能です。 デフォルト値: **VirusBase** = %var_dir/bases/*.vdb
UpdatePath = {path to directory}	Updater (update.pl) の一時作業用ディレクトリの指定です。 デフォルト値: **UpdatePath** = %var_dir/updates/
TempPath = {path to directory}	ウイルス検査エンジンの一時作業用ディレクトリの指定です。 デフォルト値: **TempPath** = /tmp
LngFileName = {path to file}	言語ファイルの指定です。 デフォルト値: **LngFileName** = %bin_dir/lib/

	ru_scanner.dwl
Key = {path to file}	ライセンスキーファイルの指定です。 デフォルト値: **Key** = %bin_dir/drweb32.key
OutputMode = {Terminal | Quiet}	drwebプロセスの起動時のメッセージ出力の指定です。 Terminal - 標準出力 Quiet - 出力を抑制 デフォルト値: **OutputMode** = Terminal
HeuristicAnalysis = {Yes | No}	未知のウイルスを検出するためのヒューリスティック解析の有効・無効の指定です。 ヒューリスティック解析を使用することで、ウイルス情報データベースに登録されていない未知のウイルスの検出に効果を発揮します。一方で、ウイルスに似たコードを持つプログラムなどを誤検出する可能性があることに留意が必要です。 デフォルト値: **HeuristicAnalysis** = Yes
ScanPriority = {value}	Scannerのプロセスの優先度の指定です。 -20 ～ 19(Linux)または、20(Linux以外)の整数 デフォルト値: **ScanPriority** = 0
FilesTypes = {ext}	**ScanFiles**パラメータがByTypeの場合に検査対象となる拡張子の指定です。 "*"と"?" によるワイルドカードの利用が可能です。 デフォルト値:

	FilesTypes = EXE, COM, SYS, OV?, BAT, BIN, DRV, PRG, BOO, SCR, CMD, VXD, 386, DLL, FON, DO?, XL?, WIZ, RTF, CL*, HT*, VB*, JS*, INF, AR?, ZIP, R??, PP?, OBJ, LIB, HLP, MD?, INI, MBR, IMG, CSC, CPL, MBP, SHS, SHB, PIF, SO, CHM, REG, XML, PRC, ASP, LSP, MSO, OBD, THE*, NWS, SWF, BMP, MPP, OCX, DVB, CPY, MSG, EML
FilesTypesWarnings = {Yes | No}	検査対象外ファイルに関する警告の指定です。 **ScanFiles**パラメータがByTypeの場合、検査対象外ファイルの検査要求に対して警告するかの指定です。 デフォルト値: **FilesTypesWarnings** = Yes
ScanFiles = {All | ByType}	検査モードの指定です。 All を指定した場合は、全てのファイルを検査します。ByTypeを指定した場合は、**FileType**パラメータで指定された拡張子のファイルのみを検査します。 デフォルト値: **ScanFiles** = All
ScanSubDirectories = {Yes | No}	サブディレクトリの検査に関する指定です。 検査対象ディレクトリ内のサブディレクトリを検査します。 デフォルト値: **ScanSubDirectories** = Yes
CheckArchives = {Yes | No}	アーカイブファイルの検査に関する指定です。

	ZIP, RAR, ARJ, TAR, GZIP, CAB その他のアーカイブファイルを検査します。 デフォルト値: **CheckArchives** = Yes
CheckEMailFiles = {Yes | No}	電子メール書式ファイルの検査に関する指定です。 デフォルト値: **CheckEMailFiles** = Yes
ExcludePaths = {path to directory}	検査を除外するディレクトリの指定です。 デフォルト値: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {Yes | No}	シンボリックリンク先のファイル・ディレクトリの検査に関する指定です。 デフォルト値: **FollowLinks** = No
RenameFilesTo = {mask}	名前変更時の拡張子のマスクに関する指定です。 デフォルト値の"#??"の場合、"#"は拡張子の該当箇所を"#"で置き換えることを意味し、"??"は該当箇所を置き換えないことを意味します。eicar.comの検出で名前変更をした場合、eicar.#omとなります。拡張子がないファイルの場合は、".#"を付加します。 デフォルト値: **RenameFilesTo** = #??
MoveFilesTo = {path to directory}	隔離先のディレクトリの指定です。 デフォルト値: **MoveFilesTo** = %var_dir/infected

EnableDeleteArchiveAction = {Yes | No}	感染ファイルを含むmultipartオブジェクト(アーカイブ, メールボックス, html)の削除に関する指定です。 このオプションを有効にした場合、感染ファイルを含むアーカイブやメールボックス(mbox形式の場合)ごと削除されますので注意してください。 デフォルト値: **EnableDeleteArchiveAction** = No
InfectedFiles = {Report | Cure | Delete | Move | Rename | Ignore}	感染ファイルに対する処理の指定です。 • Report - ログに記録 • Cure - 修復 • Delete - 削除 • Move - 隔離 • Rename - 名前変更 • Ignore - 無視 Delete, Move, Renameの処理は、感染ファイルを含むアーカイブやメールボックスの場合、ファイルごと削除されますので注意してください。 デフォルト値: **InfectedFiles** = Report
SuspiciousFiles = {Report | Delete | Move | Rename | Ignore}	感染が疑われるファイルに対する処理の指定です。 • Report - ログに記録 • Delete - 削除 • Move - 隔離 • Rename - 名前変更 • Ignore - 無視 デフォルト値: **SuspiciousFiles** = Report

IncurableFiles = {Report |Delete | Move | Rename | Ignore}	修復不可能なファイルに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **IncurableFiles** = Report
ActionAdware = {Report | Delete | Move | Rename | Ignore}	アドウェアに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionAdware** = Report
ActionDialers = {Report | Delete | Move | Rename | Ignore}	ダイヤラーに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionDialers** = Report
ActionJokes = {Report | Delete | Move | Rename | Ignore}	ジョークプログラムに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離

	• Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionJokes** = Report
ActionRiskware = {Report | Delete | Move | Rename | Ignore}	リスクウェアに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionRiskware** = Report
ActionHacktools = {Report | Delete | Move | Rename | Ignore}	ハッキングプログラムに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionHacktools** = Report
ActionInfectedMail = {Report | Delete | Move | Rename | Ignore}	感染ファイルを含むメールボックスに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値:

	ActionInfectedMail = Report
ActionInfectedArchive = {Report | Delete | Move | Rename | Ignore}	感染ファイルを含むアーカイブに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionInfectedArchive** = Report
ActionInfectedContainer = {Report | Delete | Move | Rename | Ignore}	感染ファイルを含むコンテナに対する処理の指定です。 • Report – ログに記録 • Delete – 削除 • Move – 隔離 • Rename – 名前変更 • Ignore – 無視 デフォルト値: **ActionInfectedContainer** = Report
LogFileName = {filename or syslog}	ログファイルの指定です。 syslogを指定することができます。(**SyslogFacility**と**SyslogPriority**パラメータの指定が必要です。) デフォルト値: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern |	syslogのファシリティの指定です。 デフォルト値: **SyslogFacility** = User

User | Mail}	
SyslogPriority = {Alert | Warning | Notice | Info | Error}	syslogのプライオリティの指定です。 デフォルト値: **SyslogPriority** = Info
LimitLog = {Yes | No}	ログファイルのサイズ制限の指定です。**LogFileName** = syslogの場合は無視されます。 有効(yes)にすると**Scanner**起動時にログファイルのサイズをチェックし、最大サイズを超えている場合にログファイルを削除します。(直近のログファイルは、.bakの拡張子で残されます。)ログファイルの最大サイズは、**MaxLogSize**パラメータで指定します。 デフォルト値: **LimitLog** = No
MaxLogSize = {value}	ログファイルの最大サイズの指定です。(**LimitLog** = Yes の場合) 0以上の整数で、ログファイルのサイズ(キロバイト)を指定します。 デフォルト値: **MaxLogSize** = 512
LogScanned = {Yes | No}	Yesの場合、検査した全てのファイルの情報をログに記録します。 デフォルト値: **LogScanned** = Yes
LogPacked = {Yes | No}	Yesの場合、DIET, PKLITE などのパッカ? に関する情報をログに記録します。 デフォルト値: **LogPacked** = Yes

LogArchived = {Yes | No}	Yesの場合、アーカイバに関する情報をログに記録します。 デフォルト値: **LogArchived** = Yes
LogTime = {Yes | No}	Yesの場合、ログの各行に処理時間を記録します。(**LogFileName** = syslogの場合は使用できません。) デフォルト値: **LogTime** = Yes
LogStatistics = {Yes | No}	Yesの場合、検査の統計情報をログに記録します。 デフォルト値: **LogStatistics** = Yes
RecodeNonprintable = {Yes | No}	ログ中の表示できない文字の置換に関する指定です。 デフォルト値: **RecodeNonprintable** = Yes
RecodeMode = {Replace | QuotedPrintable}	表示できない文字の置換方法の指定です。(**RecodeNonprintable** = Yesの場合) **Replace**の場合は、**RecodeChar**パラメータで指定する文字に置換します。 QuotedPrintableの場合は、quoted-printableエンコード文字列で置換します。 デフォルト値: **RecodeMode** = QuotedPrintable
RecodeChar = {"?" | "_" | ...}	表示できない文字を置換する文字列の指定です。(**RecodeMode** = Replaceの場合) デフォルト値:

	RecodeChar = "?"

アーカイブファイルの検査時間を短縮するために次のパラメータを使用することができます。

MaxCompressionRatio = {value}	アーカイブファイルを展開して検査する際の圧縮比の上限値の指定です。 指定された圧縮比を超える場合には検査を行いません。
	デフォルト値: **MaxCompressionRatio** = 5000
CompressionCheckThreshold = {value}	アーカイブファイルの圧縮比を確認するファイルサイズの下限値(キロバイト)の指定です。
	デフォルト値: **CompressionCheckThreshold** = 500000
MaxFileSizeToExtract = {value}	アーカイブ中の最大ファイルサイズ(キロバイト)の指定です。 指定された値を超えたファイルは検査を行いません。
	デフォルト値: **MaxFileSizeToExtract** = 500000
MaxArchiveLevel = {value}	アーカイブファイルを検査する際の最大ネストレベルの指定です。 最大ネストレベルを超えるアーカイブファイルは検査を行いません。
	デフォルト値: **MaxArchiveLevel** = 8

MaximumMemoryAllocationSize = {value in Mbytes}	ファイルを検査する際に消費するメモリについて、最大値を制限するための指定です。(メガバイト) "0"が指定された場合、制限はありません。 デフォルト値: **MaximumMemoryAllocationSize** = 0
ScannerScanTimeout = {time in seconds}	ファイル検査のタイムアウト値の指定です。(秒) "0"が指定された場合、タイムアウトはしません。 デフォルト値: **ScannerScanTimeout** = 0
MaxBasesObsolescencePeriod = {time in hours}	ウイルス定義ファイルが古くなっていないかを示すための期間です。(時間) 最終更新から指定された期間を経過すると、ウイルス定義ファイルが古くなっていることを示す通知がコンソールに出力されます。"0"が指定された場合、チェックされません。 デフォルト値: **MaxBasesObsolescencePeriod** = 24
ControlAgent = {Agent socket address}	**Agent**のソケットの指定です。 localまたは、inet - TCPソケット, unix - UNIXソケット 例: **ControlAgent** = inet:4040@127.0.0.1,local:%var_dir/ipc/.agent デフォルト値: **ControlAgent** = local:%var_dir/ipc/.agent

OnlyKey = {Yes | No}	Yesの場合、Scannerは、Agentからライセンスキーファイルのみを受け取り、ローカルドライブ上の設定ファイルを使用します。 Noの場合、Scannerは、Agentからライセンスキーファイルと設定情報を受け取ります。 デフォルト値: **OnlyKey** = No

Dr.Web Scannerの起動

Dr.Web Scannerは以下のコマンドで実行します。

```
$ %bin_dir/drweb
```

%bin_dir ディレクトリがPATH環境変数に追加されている場合、どこからでも"drweb"とコマンド入力するだけでDr.Web Scannerを実行することができます。

Dr.Web Scannerは、root権限でもユーザ権限でも実行することができます。ユーザ権限の場合は、アクセス権の関係などによる制限を受ける場合があります。(ウイルス検出時の隔離や名前変更など)

Scannerが実行されると、プログラムバージョンのほか、ウイルス定義ファイルやライセンスキーに関する情報などを出力します。

```
Dr.Web (R) Scanner for Linux, v6.0.1 (February 19, 2010)
Copyright (c) Igor Daniloff, 1992-2010
Support service: http://support.drweb.com/
To purchase: http://buy.drweb.com/
Program version: 6.0.1.10060 <API:2.2>
Engine version: 6.0.1.9170 <API:2.2>
Loading /var/drweb/bases/drwtoday.vdb - Ok, virus records: 1533
```

```
Loading  /var/drweb/bases/drw60012.vdb  -  Ok,
virus records: 3511
----------------------------------------
Loading  /var/drweb/bases/drw60000.vdb  -  Ok,
virus records: 1194
Loading  /var/drweb/bases/dwn60001.vdb  -  Ok,
virus records: 840
Loading  /var/drweb/bases/drwebase.vdb  -  Ok,
virus records: 78674
Loading  /var/drweb/bases/drwrisky.vdb  -  Ok,
virus records: 1271
Loading  /var/drweb/bases/drwnasty.vdb  -  Ok,
virus records: 4867
Total virus records: 538681
Key file: /opt/drweb/drweb32.key
Key file number: XXXXXXXXXX
Key file activation date: XXXX-XX-XX
Key file expiration date: XXXX-XX-XX
```

Scannerが終了すると、レポートが表示されます。ウイルス検出時の動作を指定する場合は、コマンドラインパラメータを追加します。

Dr.Web Daemon

Dr.Web Daemonは、他の**Dr.Web**コンポーネントからの検査要求に応じ、ウイルス検査を行うアンチウイルスモジュールです。ディスク上のファイルまたは、ソケット経由で転送されたデータの両方を検査することが可能です。**Dr.Web Daemon**によって、ウイルス定義ファイルに登録されている既知のウイルスを検出、修復することができます。(修復できない場合についても、削除・隔離等の処理が行えます。)

Dr.Web Daemonは、検査要求を受け付けるためにデーモンとして常時動作しています。

Dr.Web for Unix Internet gatewaysは、**Dr.Web ICAP**と**Dr.Web Daemon**がICAPプロトコルに対応したアプリケーションと連携するアンチウイルスソリューションです。

コマンドラインパラメータ

Dr.Web Daemonは、コマンラインパラメータの使用をサポートしています。パラメータは、ハイフン"-"で指定し、スペースで区切ります。

パラメータのヘルプを表示する場合は、以下のように`-?`, `-h` または、`-help`を指定します。

```
$ drwebd -h
```

Dr.Web Daemonでは、以下のコマンドラインパラメータを使用することができます。

- `-ini=<path to file>` – 設定ファイルへのパスの指定です。(デフォルト以外を指定する場合)
- `--foreground=<yes|no>` – **Daemon**の動作モードの指定です。"`Yes`"の場合、**Daemon**はフォアグラウンドモードで動作し、"`No`"の場合はバックグラウンド(デーモン)モードで動作します。
- `--check-only <command line parameters for checking>` – 設定ファイルと指定されたコマンドラインパラメータのチェックを行います。

- -a=<Agent address> – Daemonを集中管理モードで動作させます。(設定ファイル、ライセンスキーファイルをAgentから受け取ります。)
- --only-key – Agentからライセンスキーファイルのみを受け取ります。(ローカルの設定ファイルを使用します。)

設定

Dr.Web Daemonの設定は、要件や状況に応じて設定を変更することができます。Daemonの設定は、設定ファイルに保存されています。(デフォルト: %etc_dir/drweb32.ini)

別の設定ファイルを使用する場合は、コマンドラインパラメータの -ini パラメータを使用し、設定ファイルをフルパスで指定してください。

[Daemon] section

EnginePath = {path to file}	ウイルス検査エンジン(drweb32.dll)の指定です。 デフォルト値: **EnginePath** = %var_dir/lib/drweb32.dll
VirusBase = {list of files or masks}	ウイルス定義ファイルの指定です。 ワイルドカード"*"の利用とカンマ","区切りによる複数指定が可能です。 デフォルト値: **VirusBase** = %var_dir/bases/*.vdb
UpdatePath = {path to directory}	Updater(update.pl)の一時作業用ディレクトリの指定です。 デフォルト値: **UpdatePath** = %var_dir/updates/

TempPath = {path to directory}	ウイルス検査エンジンの一時作業用ディレクトリの指定です。 **TempPath** = %var_dir/spool/
Key = {path to file}	ライセンスキーファイルの指定です。 デフォルト値: **Key** = %bin_dir/drweb32.key
MailAddressesList = {path to file}	検査対象とするメールアドレス一覧ファイルへのパスの指定です。15または30アドレスライセンスの場合に有効となります。 デフォルト値: **MailAddressesList** = %etc_dir/email.ini
OutputMode = {Terminal | Quiet}	drwebプロセスの起動時のメッセージ出力の指定です。 Terminal - 標準出力 Quiet - 出力を抑制 デフォルト値: **OutputMode** = Terminal
RunForeground = {Yes | No}	Yesの場合、**Daemon**をフォアグラウンドで動作させます。 通常は、デフォルトのNo(デーモン)を使用します。特別なユーティリティ(daemontoolsなど)を使用する場合の指定です。 デフォルト値: **RunForeground** = No
User = {user name}	**Daemon**を起動するユーザ名の指定です。 デフォルト値: **User** = drweb

PidFile = {path to file}	**Daemon**のPIDファイルへのパスの指定です。 デフォルト値: **PidFile** = %var_dir/run/drwebd.pid
BusyFile = {path to file}	**Daemon**の子プロセスが検査中に作成するロックファイルへのパスの指定です。 ファイル名の末尾に子プロセスのPIDが付加されます。(例:/var/run/drwebd.bsy.123456) デフォルト値: **BusyFile** = %var_dir/run/drwebd.bsy
ProcessesPool = {process pool settings}	子プロセスの生成に関する指定です。 • auto - システムの負荷状態により、自動的にプロセス数が生成されます。 • N - 1以上の整数を指定します。指定した数だけ、プロセスが生成され、必要に応じて追加プロセスが生成されます。 • N-M - 1以上の整数を指定します。Nに指定した数だけ、プロセスが予め生成されますが、Mで指定された数を超えてプロセスは生成されません。 • **timeout** = {time in seconds} - アクティブでないプロセスを終了するまでのタイムアウト値の指定です。(前述のNで指定された数のプロセスは、このパラメータの影響を受けません。) • **stop_timeout** = {time in seconds} - 稼働プロセスを停止させるまでの最大待ち時間の指定です。 デフォルト値: **ProcessesPool** = auto,**timeout** = 120, **stat** = no, **stop_timeout** =

	1
OnlyKey = {Yes | No}	Yesの場合、Agentからライセンスキーファイルのみを受け取ります。設定は、ローカルの設定ファイルに従います。 Noの場合、Agentからライセンスキーファイルと設定ファイルを受け取ります。 デフォルト値: **OnlyKey** = No
ControlAgent = {socket address}	Agentのソケットアドレスの指定です。 • inet - TCPソケット • local または、unix - UNIXソケット 例: **ControlAgent** = inet:4040@127.0.0.1,local:%var_dir/ipc/.agent デフォルト値: **ControlAgent** = local:%var_dir/ipc/.agent
MailCommand = {command}	DaemonおよびUpdaterが通知メールを管理者に送信する際のコマンドの指定です。 デフォルト値: **MailCommand** = "/usr/sbin/sendmail -i -bm -f drweb -- root"
NotifyPeriod = {value}	ライセンスキーの期限切れを示す通知メールを何日前から送信するかの指定です。"0"を指定した場合、ライセンスの期限が切れた後に通知メールが送信されます。 デフォルト値: **NotifyPeriod** = 14

NotifyFile = {path to file}	ライセンスキーの期限切れを示す通知メールを送信した日時を記録するファイルの指定です。 デフォルト: **NotifyFile** = %var_dir/.notify
NotifyType = {Ever | Everyday | Once}	ライセンスキーの期限切れを示す通知メールを送信する頻度の指定です。 Once - 1回かぎり Everyday - 毎日 Ever - Daemonの再起動時およびウイルス定義ファイルの更新時 デフォルト値: **NotifyType** = Ever
FileTimeout = {value in seconds}	1個のファイルを検査する最大時間(秒)の指定です。 デフォルト値: **FileTimeout** = 30
StopOnFirstInfected = {Yes | No}	ウイルスを1個検出した時点で検査を終了する場合の指定です。 デフォルト値: **StopOnFirstInfected** = No
ScanPriority = {value}	Daemonのプロセスの優先度の指定です。 -20 ~ 19 (Linux) または、20 (Linux以外)の整数 デフォルト値: **ScanPriority** = 0
FilesTypes = {list of file extensions}	**ScanFiles**パラメータがByTypeの場合に検査対象となる拡張子の指定です。 "*" と "?" によるワイルドカードの利用が可能

	です。 デフォルト値: **FilesTypes** = EXE, COM, SYS, OV?, BAT, BIN, DRV, PRG, BOO, SCR, CMD, VXD, 386, DLL, FON, DO?, XL?, WIZ, RTF, CL*, HT*, VB*, JS*, INF, AR?, ZIP, R??, PP?, OBJ, LIB, HLP, MD?, INI, MBR, IMG, CSC, CPL, MBP, SHS, SHB, PIF, SO, CHM, REG, XML, PRC, ASP, LSP, MSO, OBD, THE*, NWS, SWF, BMP, MPP, OCX, DVB, CPY, MSG, EML
FilesTypesWarnings = {Yes | No}	検査対象外ファイルに関する警告の指定です。 **ScanFiles**パラメータがByTypeの場合、検査対象外ファイルの検査要求に対して警告するかの指定です。 デフォルト値: **FilesTypesWarnings** = Yes
ScanFiles = {All | ByType}	検査モードの指定です。 All を指定した場合は、全てのファイルを検査します。ByTypeを指定した場合は、**FileType**パラメータで指定された拡張子のファイルのみを検査します。 デフォルト値: **ScanFiles** = All
CheckArchives = {Yes | No}	アーカイブファイルの検査に関する指定です。 ZIP, RAR, ARJ, TAR, GZIP, CAB その他のアーカイブファイルを検査します。 デフォルト値: **CheckArchives** = Yes

CheckEMailFiles = {Yes | No}	電子メール書式ファイルの検査に関する指定です。 デフォルト値: **CheckEMailFiles** = Yes
ExcludePaths = {path}	検査を除外するディレクトリの指定です。 デフォルト値: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {Yes | No}	シンボリックリンク先のファイル・ディレクトリの検査に関する指定です。 デフォルト値: **FollowLinks** = No
RenameFilesTo = {mask}	名前変更時の拡張子のマスクに関する指定です。 デフォルト値の"#??"の場合、"#"は拡張子の該当箇所を"#"で置き換えることを意味し、"??"は該当箇所を置き換えないことを意味します。eicar.comの検出で名前変更をした場合、eicar.#omとなります。拡張子がないファイルの場合は、".#"を付加します。 デフォルト値: **RenameFilesTo** = #??
MoveFilesTo = {path to directory}	隔離先のディレクトリの指定です。 デフォルト値: **MoveFilesTo** = %var_dir/infected/
BackupFilesTo = {path to directory}	感染ファイルに対して修復(Cure)を指定している場合に、元のファイルをバックアップするディレクトリの指定です。. デフォルト値:

	BackupFilesTo = %var_dir/infected/
LogFileName = {filename or syslog}	ログファイルの指定です。 syslogを指定することができます。(**SyslogFacility**と**SyslogPriority**パラメータの指定が必要です。) デフォルト値: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	syslogのファシリティの指定です。 デフォルト値: **SyslogFacility** = Daemon
SyslogPriority = {Alert | Warning | Notice | Info | Error}	syslogのプライオリティの指定です。 デフォルト値: **SyslogPriority** = Info
LimitLog = {Yes | No}	ログファイルのサイズ制限の指定です。 デフォルト値: **LimitLog** = No
MaxLogSize = {value in Kbytes}	ログファイルの最大サイズの指定です。(**LimitLog** = Yes の場合) 0以上の整数で、ログファイルのサイズ(キロバイト)を指定します。 デフォルト値: **MaxLogSize** = 512
LogScanned = {Yes | No}	Yesの場合、検査した全てのファイルの情報をログに記録します。 デフォルト値: **LogScanned** = Yes

LogPacked = {Yes | No}	Yesの場合、DIET, PKLITEなどのパッカ? に関する情報をログに記録します。 デフォルト値: **LogPacked** = Yes
LogArchived = {Yes | No}	Yesの場合、アーカイバに関する情報をログに記録します。 デフォルト値: **LogArchived** = Yes
LogTime = {Yes | No}	Yesの場合、ログの各行に処理時間を記録します。(**LogFileName** = syslogの場合は使用できません。) デフォルト値: **LogTime** = Yes
LogProcessInfo = {Yes | No}	Yesの場合、検査を実施したプロセスのPIDと検査を要求したクライアントのIPアドレス(または、ホスト名)が記録されます。 デフォルト値: **LogProcessInfo** = Yes
RecodeNonprintable = {Yes | No}	ログ中の表示できない文字の置換に関する指定です。 デフォルト値: **RecodeNonprintable** = Yes
RecodeMode = {Replace | QuotedPrintable}	表示できない文字の置換方法の指定です。(**RecodeNonprintable** = Yesの場合) **Replace**の場合は、**RecodeChar**パラメータで指定する文字に置換します。 QuotedPrintableの場合は、quoted-printableエンコード文字列で置換します。 デフォルト値:

	RecodeMode = QuotedPrintable
RecodeChar = {"?" | "_" | ...}	表示できない文字を置換する文字列の指定です。(**RecodeMode** = Replaceの場合) デフォルト値: **RecodeChar** = "?"
Socket = {socket address}	デーモンが検査要求を待ちうけるソケットの指定です。 • inet - TCPソケット • local または、unix - UNIXソケット 例: Socket = inet:3000@127.0.0.1,local:%var_dir/.drwebd 例: Socket = 3000 127.0.0.1, 192.168.0.100 例: Socket = %var_dir/.drwebd 0660 Socketパラメータの数に制限はありません。正しく指定されたすべての設定でデーモンが動作します。利用可能なすべてのインターフェースで検査要求を受ける場合は、3000 0.0.0.0 と指定してください。 デフォルト値: **Socket** = %var_dir/run/.daemon **Socket** = 3000, localhost
SocketTimeout = {value in seconds}	ソケット経由で送受信されるデータのタイムアウト値(秒)の指定です。ファイルの検査時間は含みません。 デフォルト値: **SocketTimeout** = 10

アーカイブファイルの検査時間を短縮するために次のパラメータを使用することができます。

MaxCompressionRatio = {value}	アーカイブファイルを展開して検査する際の圧縮比の上限値の指定です。 指定された圧縮比を超える場合には検査を行いません。 デフォルト値: **MaxCompressionRatio** = 500
CompressionCheckThreshold = {value in Kbytes}	アーカイブファイルの圧縮比を確認するファイルサイズの下限値(キロバイト)の指定です。 デフォルト値: **CompressionCheckThreshold** = 1024
MaxFileSizeToExtract = {value in Kbytes}	アーカイブ中の最大ファイルサイズ(キロバイト)の指定です。 指定された値を超えたファイルは検査を行いません。 デフォルト値: **MaxFileSizeToExtract** = 40960
MaxArchiveLevel = {value}	アーカイブファイルを検査する際の最大ネストレベルの指定です。 最大ネストレベルを超えるアーカイブファイルは検査を行いません。 デフォルト値: **MaxArchiveLevel** = 8

Dr.Web Daemonの起動

Daemonがデフォルト設定で起動すると、以下の処理が実行されます。

- 設定ファイルを検索して読み込みます。設定ファイルが見つからない場合、プロセスは終了します。設定ファイルは、起動コマンドの`-ini`パラメータで指定するか、デフォルトの設定ファイル(`%etc_dir/drweb32.ini`)を使用します。
- ログファイルを作成します。**Daemon**が使用するユーザアカウントは、ログファイルディレクトリに対して適切な権限を持っている必要があります。ユーザ権限の場合、デフォルトのログディレクトリ(`/var/log/`)に対して書き込み権限がありません。**User**パラメータを指定した場合、**LogFileName**パラメータでログファイルを適切な場所に指定する必要があります。
- 設定ファイルで指定された場所からライセンスキーファイルを読み込みます。ライセンスキーファイルが見つからない場合、プロセスは終了します。
- **User**パラメータが指定された場合、**Daemon**はユーザアカウントを作成し、適切な権限を付与します。(デフォルト値:`drweb`)
- **Engine**(`drweb32.dll`)を読み込みます。**Engine**が見つからない場合、または異常な場合、プロセスは終了します。
- ウイルス定義ファイルを読み込みます。
- `daemon`モードになり、ログファイルに情報を出力します。
- **Daemon**と他の**Dr.Web for Unix Internet gateways**モジュールが連携するためのソケットを作成します。ソケットが作成できない場合、プロセスは終了します。
- **Daemon**のPIDファイルが作成されます。PIDファイルが作成できない場合、プロセスは終了します。

シグナルの処理

Dr.Web Daemonは、以下のシグナルを受け取ることができます。

- `SIGHUP` – 設定ファイルの再読み込み
- `SIGTERM` – **Daemon**の終了要求
- `SIGKILL` – **Daemon**の強制終了

Dr.Web Daemonのテストと診断

Daemonの稼働状態を確認する場合、以下のようなコマンドを実行します。

```
$ netstat -a
```

必要なソケットが作成されているかを確認します。

TCP ソケット:

```
--- cut ---
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 localhost:3000 *:* LISTEN
raw 0 0 *:icmp *:* 7
raw 0 0 *:tcp *:* 7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags Type State I-Node Path
unix 0 [ ACC ] STREAM LISTENING 384 /dev/gpmctl
unix 0 [ ] STREAM CONNECTED 190 @0000001b
unix 1 [ ] STREAM CONNECTED 1091 @00000031
unix 0 [ ACC ] STREAM LISTENING 403 /tmp/.font-unix/fs7100
unix 4 [ ] DGRAM 293 /dev/log
unix 1 [ ] STREAM CONNECTED 1092 /dev/gpmctl
unix 0 [ ] DGRAM 450
unix 0 [ ] DGRAM 433
unix 0 [ ] DGRAM 416
unix 0 [ ] DGRAM 308
```

```
--- cut ---
```

UNIX ソケット:

```
--- cut ---
Active Internet connections (servers and
established)
Proto Recv-Q Send-Q Local Address Foreign
Address State
raw 0 0 *:icmp *:* 7
raw 0 0 *:tcp *:* 7
Active UNIX domain sockets (servers and
established)
Proto RefCnt Flags Type State I-Node Path
unix 0 [ ACC ] STREAM LISTENING 384 /dev/
gpmctl
unix 0 [ ] STREAM CONNECTED 190 @0000001b
unix 1 [ ] STREAM CONNECTED 1091 @00000031
unix 0 [ ACC ] STREAM LISTENING 1127 /opt/
drweb/run/drwebd.skt
unix 0 [ ACC ] STREAM LISTENING 403 /tmp/.
font-unix/fs7100
unix 4 [ ] DGRAM 293 /dev/log
unix 1 [ ] STREAM CONNECTED 1092 /dev/
gpmctl
unix 0 [ ] DGRAM 450
unix 0 [ ] DGRAM 433
unix 0 [ ] DGRAM 416
unix 0 [ ] DGRAM 308
--- cut ---
```

ソケットが作成されていない場合、**Daemon**の起動に失敗しています。

Daemonのコンソールクライアントスト機能を(drwebdc)を使用したテスト、サ

ービス情報の確認を行ってください。

TCP ソケット:

```
$ drwebdc -nHOSTNAME -pPORTNUM -sv -sb
```

UNIX ソケット:

```
$ drwebdc -uSOCKETFILE -sv -sb
```

以下のような情報が出力されます。

```
--- cut ---
- Version: DrWeb Daemon 6.00
- Loaded bases:
Base /var/drweb/bases/drwtoday.vdb contains 5 records.
Base /var/drweb/bases/drw50003.vdb contains 409 records.
Base /var/drweb/bases/drw50002.vdb contains 543 records.
Base /var/drweb/bases/drwebase.vdb contains 51982 records.
Base /var/drweb/bases/drw50001.vdb contains 364 records.
Total 53303 virus-finding records.
--- cut ---
```

上記のような出力結果を得られない場合は、診断モードでdrwebdcを実行します。

TCP ソケット:

```
$ drwebdc -nHOSTNAME -pPORTNUM -sv -sb -v
```

UNIX ソケット:

```
$ drwebdc -uSOCKETFILE -sv -sb -v
```

詳細ログを元に問題箇所の特定が行えます。

```
--- cut ---
dwlib: fd: connect() failed - Connection refused
dwlib: tcp: connecting to 127.0.0.1:3300 - failed
dwlib: cannot create connection with a DrWeb daemon
ERROR: cannot retrieve daemon version
Error -12
--- cut ---
```

Demonのテストを行う場合は、製品パッケージに含まれているreadme.eicarファイルを使用することができます。readme.eicarファイルをテキストエディタで開き、記載内容に従ってeicar.comファイルに変更してください。

TCP ソケット:

```
$ drwebdc -nHOSTNAME -pPORTNUM -f eicar.com
```

UNIX ソケット:

```
$ drwebdc -uSOCKETFILE -f eicar.com
```

以下のような情報がコンソールに出力されることを確認してください。

```
--- cut ---
Results: daemon return code 0x20
(known virus is found)
--- cut ---
```

検査モード

Dr.Web Daemonは、以下のモードでウイルス検査が可能です。

- ソケット経由で受信するデータの検査（リモート検査モード）
- ディスク上のファイル検査（ローカル検査モード）

リモート検査モードの場合、**Daemon**はソケットから検査データを受信します。**Daemon**は、ソケットから受信したあるゆるデータ、ファイルを検査することができます。

ローカル検査モードの場合、**Daemon**はディスク上の指定されたファイルの検査を行います。ローカル検査モードは、効率的で簡単に利用できるというメリットがあり、コンソールクライアントやフィルタなどのクライアントは、ファイルのパスだけを**Daemon**に送ることで検査することができます。

Dr.Web ICAP

Dr.Web ICAPモジュール(`drweb-icapd`)は、ICAPに対応したプロキシサーバと連携します。

本章では、SquidおよびSafeSquidを使用した設定を紹介しています。

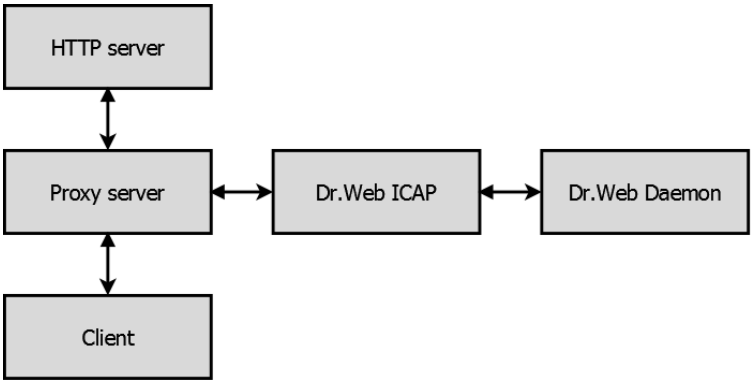
Dr.Web ICAPは、HTTPおよびFTPトラフィックのウイルス検査を行うためにプロキシサーバと**Dr.Web Daemon**との接続を確立します。

Dr.Web ICAPは、プロキシサーバからの接続を受けて、コンテンツのウイルス検査を**Dr.Web Daemon**に要求します。また、ダウンロードファイルのサイズやホスト名によるアクセス制御のほか、URLのブラックリストを参照してユーザを危険なコンテンツから保護する機能も備えています。**Dr.Web ICAP**は、インターネットゲートウェイにおけるウイルス対策で重要な役割を担います。

Dr.Web ICAPとSquidの設定

プロキシサーバと`drweb-icapd`を含めた基本的な構成は以下のとおりです。

図 15. HTTPトラフィックの検査



上の図では、クライアントがプロキシサーバを経由してどのようにHTTサーバに接続するかが示されています。

プロキシサーバは、ICAPクライアントとして動作し、**Dr.Web ICAP**は、ICAPサーバとして動作します。また、 **Dr.Web ICAP**モジュールは、**Dr.Web Daemon**のクライアントとして動作しています。**Dr.Web ICAP**は、プロキシサーバとICAPプロトコルで連携し、すべてのHTTPトラフィックを検査します。

この基本構成の場合、FTPトラフィックを検査することはできません。FTPトラフィックの検査については、後述する「Setting up FTP-traffic scanning with Squid」を参照してください。

Squidと**Dr.Web ICAP**を連携させるには、Squidの設定ファイルを編集する必要があります。(`squid.conf`)

以下の設定を確認し、必要に応じて設定ファイルの最後に値を指定してください。

ICAPを有効にします:

```
icap_enable on
```

ICAPサービスを登録します:

```
# ICAP service description:
# icap_service <name> <type> <pass> <url>
#    <name> ? name of service
#    <type> ? type of service
#    <pass> ? can content be passed (1) past
# ICAP server or not (0)
#    <url> - url of service
icap_service service_1 respmod_precache 0
icap://localhost:1344/respmod
```

サービスのクラスを作成します:

```
icap_class class_1 service_1
```

作成したクラスへのアクセスを許可します:

```
icap_access class_1 allow all
```

プレビューモード を使用する場合、追加設定を行う必要があります。

プレビューモードを有効にします:

```
icap_preview_enable on
```

送信するコンテンツのサイズ(プレビューサイズ)を指定します:

```
icap_preview_size 0
```

クライアントIP情報のロギングを有効にします:

```
icap_send_client_ip on
```

drweb-icapdとSquidのHTTP Keep-Alive(persistent connection)を有効にします:

```
icap_persistent_connections on
```

Squidには、respmod-postcache が実装されていないため、キャッシュ済みのコンテンツを検査することはできません。

Dr.Web ICAPとSafeSquidの設定

Dr.Web ICAPとSafeSquidを連携させる場合は、Webインターフェースまたは、config.xmlでSafeSquidの設定を編集します。

Webインターフェースを使用する場合、ICAPセクションから **Add** を選択し、新しいICAPインターフェースを追加します。以下の情報を参考に設定してください。

Enabled = true;

Host = はdrweb-icapdが起動しているIPアドレス、またはホスト名(デフォルトlocalhost)です;

File = /respmod;

Port = はdrweb-icapdが使用するポート番号(デフォルト1344)です;

Applies to = responses;

設定が完了したら、**Submit** ボタンを押して設定を反映します。

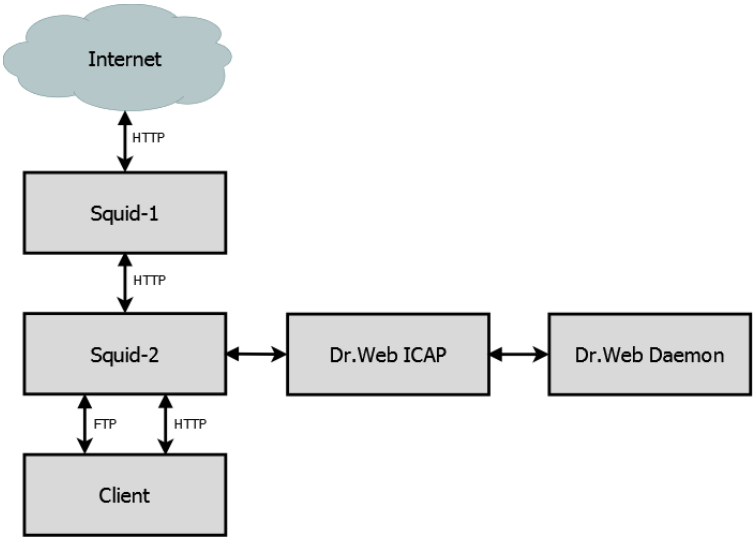
config.xml を直接編集する場合は、以下の設定を**<safesquid>**セクションに追加します。

```
<icap>
    <enabled>true</enabled>
    <icap>
        <enabled>true</enabled>
        <comment>Dr.Web icap server</comment>
        <profiles></profiles>
        <host>localhost</host>
        <file>/respmod</file>
        <port>1344</port>
        <which>responses</which>
    </icap>
</icap>
```

SquidでのFTPトラフィックスキャンの設定

Squidを多段構成にすることで、drweb-icapd によるFTPトラフィックの検査を行うことができます。

図 16. HTTPおよびFTPトラフィックの検査



図の構成では、クライアントからのHTTPおよびFTPトラフィックを **Squid-2** が受け付け、**Squid-1** に中継します。
Squid-2 が **Squid-1** からHTTPで結果を受け取ることで、FTPおよびHTTPトラフィックの両方を検査できるようになります。

Squidを多段構成にするために以下の手順を実施します。

1. 2つのSquidを別々のディレクトリにインストールします。
2. **Squid-1** の **http_port** を変更します。(例:3129)
3. **Squid-2** がdrweb-icapdと連携するように設定を編集します。
4. **Squid-2** が **Squid-1** を上位サーバとして使用するように、**Squid-2** の設定を編集します。

```
cache_peer localhost parent 3129 3130
default connect-timeout=80000
```

31299は、**Squid-1** のポート番号であり、localhost **Squid-1** がインストールされているホストの指定です。また、80000 は **Squid-2** と **Squid-1** の接続を維持するタイムアウト値の指定です。

5. Squidとdrweb-icapdを起動します。
6. HTTPおよびFTPトラフィックが **Squid-2** を経由するようにクライアントの設定を行います。(プロキシの指定)

プレビューモード

プレビューモードを使用することで、動画や音声データなど検査を行わないファイルを指定することができます。

Allowパラメータと**preview_size**パラメータ(**Allow** 204と **icap_preview_size** 0)によって、データ転送を最小限に抑え、検査時間の短縮が可能です。

Squid 2.xを使用している場合は、drweb-icapd、Squidともにプレビューモードは無効にすることを推奨します。

```
drweb-icapd:    UsePreview = No
Squid:          icap_preview_size -1
```

また、SafeSquidでは、プレビューモードの代わりにダウンロードデータの総量に関する統計のみを得ることができます。

コンテンツブラックリスト

Dr.Web ICAPは、URLフィルタリング(ブラックリスト)によるアクセス制限をサポートしています。

URLフィルタリングのブラックリストは、以下の11種類に分類されており、ウイルス定義ファイルと同様に**Dr.Web Updater**によって自動的に更新されます。

- **Porno** - ポルノ関連サイト

- **Violence** - 暴力的なサイト
- **Weapon** - 武器・兵器関連サイト
- **Gamble** - ギャンブル関連サイト
- **Drugs** - 薬物関連サイト
- **Obscenity** - 不快な内容を含むサイト
- **Chats** - チャット関連サイト
- **Terrorism** - テロ関連サイト
- **Email** - フリーメール関連サイト
- **SocialNetwork** - SNSサイト
- **MalwareLinks** - マルウェア関連サイト

各ブラックリストは、**BlockNAME**パラメータで個別に選択して使用することができます。また、コンテンツフィルタリングのブラックリストは、`*.dws` の拡張子を持っています。

コマンドラインパラメータ

`drweb-icapd`では、以下のコマンドラインパラメータを使用することができます。

- `-h` または、 `--help` - コマンドラインパラメータのヘルプを表示します。
- `-v` または、 `--version` - `drweb-icapd` のバージョンを表示します。
- `-d` - 詳細ログをコンソールに出力します。
- `-f <path to file | path to Agent socket>` - 設定ファイルへのパスの指定（デフォルト以外を指定する場合）、または**Agent**のソケットの指定（**Agent**から設定ファイルを受け取る場合）です。
- `-m` - `drweb-icapd`を**Dr.Web Monitor**のもとで動作させます。

設定

Dr.Web ICAPの設定は、要件や状況に応じて設定を変更することができます。

Dr.Web ICAPの設定は、設定ファイルに保存されています。(デフォルト: `%etc_dir/drweb-icapd.ini`)

Logfile = {file name or syslog}	ログファイルの指定です。 syslogを指定することができます。(**SyslogFacility**と**SyslogPriority**パラメータの指定が必要です。) デフォルト値: **Logfile** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	syslogのファシリティの指定です。 デフォルト値: **SyslogFacility** = Daemon
SyslogPriority = {Alert | Warning | Notice | Info}	syslogのプライオリティの指定です。 デフォルト値: **SyslogPriority** = Info
Loglevel = {numeric value}	ログの詳細レベルの指定です。 • -1 - ログを出力しない • 0 - エラー、検出したウイルスに関するログを出力 • 1 - INFOレベルのログを出力(サービス情報やクリーンなファイルに関する情報) • 2 - 一般的なログを出力 • 4 - チャンク解析メッセージを出力 • 8 - チャンク拡張メッセージを出力 • 16 - 構文解析ログの出力 • 32 - その他のデバッグメッセージ デフォルト値: **Loglevel** = 1

MaxLogSize = {value in Mbytes}	ログファイルの最大サイズの指定です。(**LimitLog** = Yes の場合) 0以上の整数で、ログファイルのサイズ(メガバイト)を指定します。**Daemon**の起動時にログファイルのサイズがこのパラメータで指定されている値を超えている場合、ログファイルは上書きされます。0を指定した場合、ログファイルの最大サイズの制限はありません。 デフォルト値: **MaxLogSize** = 1m
Hostmaster = {address}	管理者の電子メールアドレスの指定です。 デフォルト値: **Hostmaster** = root@localhost
Infected = {Cure | Move | Truncate | Report}	感染ファイルに対する処理の指定です。 • Cure - 修復 • Move - 隔離(HTMLページでユーザにメッセージを通知) • Truncate - 削除(空ファイルをユーザに送信) • Report - レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Infected** = Cure
Incurable = {Move | Truncate | Report}	修復不可能なファイルに対する処理の指定です。 • Move - 隔離(HTMLページでユーザにメッセージを通知) • Truncate - 削除(空ファイルをユーザに送信) • Report - レポート(HTMLページでユーザにメッセージを通知)

	デフォルト値: **Incurable** = Report
Suspicious = {Move | Pass | Truncate | Report}	感染の疑いがあるファイルに対する処理の指定です。 • Pass - 無視 • Move - 隔離(HTMLページでユーザにメッセージを通知) • Truncate - 削除(空ファイルをユーザに送信) • Report - レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Suspicious** = Report
Adware = {Move | Pass | Truncate | Report}	アドウェアに対する処理の指定です。 • Pass - 無視 • Move - 隔離(HTMLページでユーザにメッセージを通知) • Truncate - 削除(空ファイルをユーザに送信) • Report - レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Adware** = Report
Dialers = {Move | Pass | Truncate | Report}	ダイヤラに対する処理の指定です。 • Pass - 無視 • Move - 隔離(HTMLページでユーザにメッセージを通知) • Truncate - 削除(空ファイルをユーザに送信)

	• Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Dialers** = Report
Jokes = {Move | Pass | Truncate | Report}	ジョークプログラムに対する処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Jokes** = Pass
Riskware = {Move | Pass | Truncate | Report}	リスクウェアに対する処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Riskware** = Pass
Hacktools = {Move | Pass | Truncate | Report}	ハッキングプログラムに対する処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知)

	• Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **Hacktools** = Pass
ArchiveRestriction = {Move | Pass | Truncate | Report}	アーカイブの検査制限に抵触したファイルへの処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **ArchiveRestriction** = Report
DaemonError = {Move | Pass | Truncate | Report}	**Daemon** による検査処理中にエラーが発生した際の処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **DaemonError** = Report
SkipObject = {Move | Pass | Truncate | Report}	パスワード保護されたファイルなど、検査が行えないファイルに対する処理の指定です。 • Pass – 無視

	• Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **SkipObject** = Pass
LicenseError = {Move | Pass | Truncate | Report}	検査処理中にライセンスに関するエラー(ライセンス期間切れなど)が発生した際の処理の指定です。 • Pass – 無視 • Move – 隔離(HTMLページでユーザにメッセージを通知) • Truncate – 削除(空ファイルをユーザに送信) • Report – レポート(HTMLページでユーザにメッセージを通知) デフォルト値: **LicenseError** = Report
Heuristic = {Yes | No}	未知のウイルスを検出するためのヒューリスティック解析の有効・無効の指定です。 使用しない場合は、ウイルス定義ファイルに登録されている既知のウイルスのみを検出可能です。 デフォルト値: **Heuristic** = Yes
LocalScan = {Yes | No}	ローカル検査モードの指定です。Yesの場合、 **Daemon**はローカル検査モードでファイルを検査します。 デフォルト値: **LocalScan** = Yes

User = {user name}	Dr.Web ICAPを実行するユーザの指定です。Dr.Web Daemonを実行しているユーザ権限で、Dr.Web ICAPを実行することが強く推奨されます。 デフォルト値: **User** = drweb
Cache = {path to directory}	一時作業用ディレクトリの指定です。 デフォルト値: **Cache** = %var_dir/cache/
DwsDirectory = {path to directory}	コンテンツフィルタリングのブラックリスト(*.dws)が配置されているディレクトリの指定です。 デフォルト値: **DwsDirectory** = %var_dir/dws
BlockPorno = {Yes | No}	ポルノ関連サイトのブラックリストを使用します。 デフォルト値: **BlockPorno** = Yes
BlockViolence = {Yes | No}	暴力的なサイトのブラックリストを使用します。 デフォルト値: **BlockViolence** = Yes
BlockWeapon = {Yes | No}	武器・兵器関連サイトのブラックリストを使用します。 デフォルト値: **BlockWeapon** = Yes
BlockGamble = {Yes | No}	ギャンブル関連サイトのブラックリストを使用します。 デフォルト値:

	BlockGamble = Yes
BlockDrugs = {Yes | No}	薬物関連サイトのブラックリストを使用します。 デフォルト値: **BlockDrugs** = Yes
BlockObscenity = {Yes | No}	不快な内容を含むサイトのブラックリストを使用します。 デフォルト値: **BlockObscenity** = Yes
BlockChats = {Yes | No}	チャット関連サイトのブラックリストを使用します。 デフォルト値: **BlockChats** = No
BlockTerrorism = {Yes | No}	テロ関連サイトのブラックリストを使用します。 デフォルト値: **BlockTerrorism** = Yes
BlockEmail = {Yes | No}	フリーメール関連サイトのブラックリストを使用します。 デフォルト値: **BlockEmail** = No
BlockSocialNetwork = {Yes | No}	SNS関連サイトのブラックリストを使用します。 デフォルト値: **BlockSocialNetwork** = No
BlockMalwareLinks = {Yes | No}	マルウェア関連サイトのブラックリストを使用します。 デフォルト値: **BlockMalwareLinks** = Yes

BlockAll = {Yes | No}	インターネット接続をすべて拒否する場合の指定です。 デフォルト値: **BlockAll** = No
WhiteDwsFiles = {list of files}	コンテンツフィルタリングによって検出させないホストを含むファイルの指定です。(ホワイトリスト) 以下のの方法でホストを定義します: host1 host2 ... デフォルト値: **WhiteDwsFiles** =
SendUrlsWithViruses = {Yes | No}	ウイルスを検出した際に、URLとウイルスの名前を自動的にDr.Webに送信します。Dr.Web Agentのインストールが必要です。 デフォルト値: **SendUrlsWithViruses** = No
Templates = {path to directory}	メッセージ通知に使用するテンプレートが配置されているディレクトリの指定です。 デフォルト値: **Templates** = %etc_dir/templates/icapd
PidFile = {path to file}	Dr.Web ICAPのPIDファイルの指定です。 デフォルト値: **PidFile** = %var_dir/run/drweb_icapd.pid

Key = {path to file}	ライセンスキーファイルの指定です。 デフォルト値: **Key** = %bin_dir/drweb32.key
BlackHosts = {list of files}	接続を拒否するホスト一覧の指定です。ホスト名、IPアドレスで指定します。 デフォルト値: **BlackHosts** =
WhiteHosts = {list of files}	ウイルス検査を行わないホスト一覧の指定です。ウイルスの誤検出を回避する場合などで使用します。ホスト名、IPアドレスで指定します。 デフォルト値: **WhiteHosts** =
MaxBlocksize = {size in Mbytes}	drweb-icapdが割り当てるメモリブロックの最大値の指定です。(メガバイト) 空きメモリに余裕がある場合、このパラメータを調整することで性能向上を図ることができます。 デフォルト値: **MaxBlocksize** = 10m
BindPort = {port number}	drweb-icapdの待ちうけポート番号の指定です。 デフォルト値: **BindPort** = 1344
BindAddress = {host address}	drweb-icapdの待ちうけIPアドレスの指定です。 デフォルト値: **BindAddress** = 127.0.0.1

DrwebAddress = {FAMILY : ADDRESS}	**Dr.Web Daemon**のソケットの指定です。 以下のいずれかの指定を行います。 • inet - TCPソケット(待ちうけアドレス@ポート番号) • local - UNIXソケット(ソケットファイルのパス) • pid - **Daemon**のPIDファイル(PIDファイルのパス) 例: **DrwebAddress** = inet:3000@localhost **DrwebAddress** = local:/usr/local/drweb/run/drwebd.skt **DrwebAddress** = pid:/usr/local/drweb/run/drwebd.pid デフォルト値: **DrwebAddress** = pid:%var_dir/run/drwebd.pid
PathToQuarantine = {path to directory}	隔離ディレクトリのパスの指定です。 デフォルト値: **PathToQuarantine** = %var_dir/infected
QuarantineFilesMode = {access permissions}	隔離したファイルのパーミッションの指定です。 デフォルト値: **QuarantineFilesMode** = 0660
Timeout = {value in seconds}	**Daemon**との接続タイムアウトの指定です。 デフォルト値: **Timeout** = 300

SendMail = {Yes | No}	ユーザがウイルスに感染したファイルやコンテンツにアクセスした際に、**Hostmaster**パラメータで指定した管理者の電子メールアドレスに通知する場合の指定です。 デフォルト値: **SendMail** = No
SendMailDwsBlock = {Yes | No}	ユーザがコンテンツフィルタリングのブラックリストに含まれるサイトにアクセスした際に、**Hostmaster**パラメータで指定した管理者の電子メールアドレスに通知する場合の指定です。 デフォルト値: **SendMailDwsBlock** = No
MailCommand = {command}	**SendMail**パラメータがYesの場合に、管理者の電子メールアドレスに通知するために使用するコマンドの指定です。 デフォルト値: **MailCommand** = "/usr/sbin/sendmail -i -bm -f drweb -- %s"
MailCache = {numeric value}	**SendMail**パラメータがYesの場合で、同一ユーザで繰り返し検出があったときの管理者への通知間隔の指定です。 0を指定した場合は検出の度に、管理者の電子メールアドレスに通知されます。 デフォルト値: **MailCache** = 60
AclList = {list of files}	drweb-icapdへの接続を許可するIPアドレスとホスト名一覧の指定です。 指定がない場合、または一覧にエントリが存在しない場合、drweb-icapdはすべてのクライアントからの接続を受け付けます。

	デフォルト値: **AclList** =
SendStat = {Yes | No}	検出したウイルス統計を**Agent**経由で送信するための指定です。 デフォルト値: **SendStat** = No
KeepAlive = {Yes | No}	プロキシサーバとの接続に**KeepAlive**を使用するための指定です。 デフォルト値: **KeepAlive** = Yes
UsePreview = {Yes | No}	プレビューモードの指定です。 プレビューモードを使用すると、[MimeStart – MimeEnd]セクションでの検査・検査対象外の指定が可能となります。ただし、プロキシサーバがプレビューモードに対応している必要があります。 プロキシサーバがプレビューモードに対応していない場合は、Noを指定してパラメータを無効にします。 デフォルト値: **UsePreview** = Yes

ユーザグループへのパラメータ再設定

Dr.Web for Unix Internet gatewaysでは、ユーザやグループ毎にルールを定義したアクセス制限が行えます。現在の製品バージョンでは、以下のパラメータについて定義することができます。

BlockPorno

BlockViolence

BlockWeapon
BlockGamble
BlockDrugs
BlockObscenity
BlockChats
BlockTerrorism
BlockEmail
BlockSocialNetwork
BlockMalwareLinks
BlockAll

ルールは、drweb-icapd.iniの[match]セクションと[def]セクションで定義します。

変数

以下に示す変数を使用することができます。

変数名	変数のタイプ	説明
request_url	string	リクエストURL
request_username	string	プロキシサーバでの認証に使用するユーザ名 X-Client-Usernameヘッダから取得します。ヘッダが存在しない場合、変数は空行となります。
request_ip	IPアドレス、ネットマスク（CIDR）	プロキシサーバにリクエストを送信したユーザのIPアドレス X-Client-IPヘッダから取得します。ヘッダが存在しない場合、変数は未定義となります。
system_time	time	現在のシステム時間

論理式

論理式の構文:

```
BOOL_EXPR:
      func_name()
      COMPARE
      ( BOOL_EXPR )
      ! BOOL_EXPR
      BOOL_EXPR && BOOL_EXPR
      BOOL_EXPR || BOOL_EXPR
```

&&	AND
||	OR
!	NOT

`func_name()` はfunctionの呼び出しであり、COMPAREが以下に記載された比較処理です。functionは、予め [def] セクションで定義する必要があります。

記法	説明

`string_var`
`cidr_var`
`time_var`
	対応する変数のタイプ（文字列、CIDR、時間）

`TIME`	"HH: MM" または、"H: MM" 形式（時:分）
`STRING`	文字列
`REGEX`	POSIXの拡張正規表現
`FILE_NAME`	ファイルパス
`CIDR`	IPv4アドレス ネットマスクが指定されない場合、`/32`を意味します。

比較処理	説明
`string_var == STRING`	変数が文字列に一致
`string_var != STRING`	変数は文字列に不一致
`string_var ~ REGEX`	正規表現による検索条件に一致
`string_var == file: FILE_NAME`	変数が指定されたファイルに一致
`string_var ~ file: FILE_NAME`	変数が正規表現で指定されたファイルに一致

比較処理	説明
`cidr_var <<= CIDR`	IPアドレスが指定した範囲に属している
`cidr_var <<= file: FILE_NAME`	IPアドレスが指定されたファイルに記載されているネットワークに属している

比較処理	説明

`time_var > TIME`
`time_var >= TIME`
`time_var < TIME`
`time_var <= TIME`
	時間の比較

すべての比較処理には、優先度があります。優先度は以下のとおりです。

1. ！（NOT）

2. < (より少ない), <= (以下), > (より多い), >= (以上)
3. == (等しい), != (等しくない), ~ (一致), <<= (属している)
4. && (AND)
5. || (OR)

Functions: [def]セクション

最初に、[def]セクションでfunctionを定義します。

構文:

```
func_name_1 = { BOOL_EXPR }
```

BOOL_EXPRは、論理式です。

例:

is_localhostとlocal_ip functionを定義します。リクエスト元のIPアドレスがこれらのIPアドレス、またはファイルに記載された一覧に含まれている場合、条件に該当します。

```
[def]
is_localhost = { request_ip <<= "127.0.0.0/8" }
local_ip = {
      request_ip <<= "127.0.0.0/8"
      || request_ip <<= "192.168.0.0/16"
      || request_ip <<= "172.16.0.0/12"
      ||   request_ip   <<=   file:"/tmp/icapd/
other_local_ips.txt"
}
```

worktime() functionを定義します。 現在のシステム時間が9:30 ～ 13:00 または、14:00 ～ 18:15の場合、条件に該当します。

```
[def]
```

```
worktime = { (system_time>="9:30" &&
system_time<="13:00") || (system_time>="14:00"
&& system_time<"18:15") }
```

[match]セクション

drweb-icapd.iniの[match]セクションでルールを指定します。

構文:

```
if BOOL_EXPR {
      configuration section
}
```

BOOL_EXPRは論理式で、configuration sectionはパラメータのリストです。

例:

指定した時間内、IPアドレスについて、**Porno**および**Email**に関連したサイトの閲覧を拒否する場合、以下のようなルールを作成します。

```
[match]
if (local_ip() || request_ip <<=
"87.249.57.20") && worktime() {
      BlockPorno = yes
      BlockEmail = yes
}
```

例:

23:00 ～ 8:00 までの間、指定したIPアドレスについて、**Terrorism**に関連したサイトの閲覧を拒否する場合、以下のようなルールを作成します。

```
[match]
if (request_ip <<= "93.185.182.46" ||
request_ip <<= "195.98.93.66")
      && (system_time>="23:00" ||
system_time<="8:00")
```

```
{
      BlockTerrorism = yes
}
```

例:

"edx"ユーザについて、時間外(worktime()に一致しない)のインターネット接続を拒否する場合、以下のようなルールを作成します。

```
[match]
if request_username=="edx" && !worktime()
{
      BlockAll = yes
}
```

例:

指定したユーザ(正規表現または、一覧の条件に一致)について、指定URLへの接続を拒否する場合、以下のようなルールを作成します。

```
[match]
if
      (request_username ~ "john .*"
      || request_username ~ file:"/tmp/icapd/
users_re_block.txt"
      || request_username == file:"/tmp/icapd/
users_block.txt")

      && (request_url=="http://example.com/
mega_music.mp3")
{
      BlockAll = yes
}
```

例で記載している local_ip()およびworktime()は、前述の[def]

セクションで例示した内容が適用されます。

Squidの変数処理設定

request_usernameおよびrequest_ip変数で値を取得するためには、Squidの設定を編集する必要があります。squid.confに以下の記述を追加してください。

request_ip変数を有効にする設定:

```
# request_ip
icap_send_client_ip on
```

request_username変数を有効にする設定:

```
# request_username
icap_send_client_username on
icap_client_username_header X-Client-Username
icap_client_username_encode off
```

AgentとMonitorの相互関係

drweb-icapdは、**Agent**および**Monitor**と連携することができます。

drweb-icapdは、**Agent**から設定ファイルとライセンスキーファイルを受け取ります。そのため、drweb-icapdが**Agent**の制御下にある場合は、drweb-icapd.iniファイルの**Key**パラメータは無視されます。この場合、**Agent**の構成ファイル(デフォルト:%etc_dir/agent.conf)のライセンスキーファイルへのパスが使用されます。

drweb-icapdは、検出したウイルスの情報を**Agent**に送信することができます。(drweb-icapd.iniの**SendStat**パラメータで指定している場合)

Agentは、受信した情報を**Doctor Web**のWebサイトに送信します。agent.confファイルの**UUID**パラメータに、ライセンスキーファイルのmd5チェックサムが指定されている必要があります。ウイルスの検出情報を送信、提供することはウイルス解析やアンチウイルスソフトの品質向上に貢献します。

収集された統計情報を参照するには、http://stat.drweb.com/view/md5sum/ にアクセスします。md5sumは、**UUID**パラメータの値です。

drweb-icapdの起動時に、-fパラメータで**Agent**のソケットを指定することで、**Agent**に接続することができます。

Monitorは、drweb-icapdの起動を自動化し、制御します。drweb-icapdは、**Monitor**によって起動され、設定情報を**Agent**に要求します。

Monitorによってdrweb-icapdを起動するために、**Monitor**の設定ファイル(デフォルト:%etc_dir/monitor.conf)の**RunAppList**パラメータにICAPDを指定するか、**Monitor**の起動時に-r パラメータでICAPDを指定します。

コンポーネントの起動と停止は、システムの起動スクリプトで自動起動させるのではなく、**Monitor**によって制御することが推奨されます。

起動順序

以下の順に起動することが推奨されます。

- **Dr.Web Daemon**
- **Dr.Web ICAP**
- プロキシサーバ

Dr.Web ICAPのテスト

以下のような手順で、drweb-icapdのテストを実施することができます。

1. drweb-icapd.iniファイルの**Infected**, **Suspicious**, **Incurable**パラメータで Reportを指定します。

2. http://www.eicar.org/download/eicar.com にアクセスし、ブラウザの画面に感染ファイルに関する通知メッセージが表示されることを確認します。

通知メッセージが表示されない場合は、以下の設定を確認してください。

- ブラウザの設定で、HTTPプロキシとして`drweb-icapd`と連携するプロキシサーバが指定されていることを確認します。
- `drweb-icapd.ini`ファイルの**Template**パラメータで指定された場所に、通知メッセージのテンプレートが配置されていることを確認します。（デフォルト：%etc_dir/templates/icapd/）

SquidとSafeSquid projectsのリンク

Squid project:

http://squid-cache.org/.

ICAP support for Squid:

http://squid.sourceforge.net/icap/http://squid.sourceforge.net/projects.html#icap

SafeSquid project:

http://safesquid.com/.

Dr.Web UNIX Internet gatewaysコンソール

Dr.Web for Unix Internet gateways は **Dr.Web console for UNIX Internet Gateways** を使用することでWebインターフェースによる設定が行えます。WebインターフェースはWebminのプラグインとして実装されます。

Webminの詳細については、Webminのサイトを参照してください。 (http://www.webmin.com/).

Dr.Web console for UNIX Internet Gateways は、以下のPerlモジュールを必要とします:

- `XML::Parser` – XML文書の構文解析用モジュール
- `XML::XPath` – XPath式の解析・評価用モジュール
- `CGI` – CGIモジュール
- `CGI::Carp` – HTTPDエラーレポート作成用モジュール
- `Cwd` – カレントディレクトリ名を取得するモジュール
- `Data::Dumper` – データ構造を出力するためのモジュール
- `Encode` および `Encode::Detect` – はエンコード変換のモジュールです
- `perl-devel`または、`libperl-dev`(利用しているUnix環境によって異なります。)
- `File::Basename` – は ファイル名解析のPerlモジュールです
- `File::Stat` – 組み込み関数 stat()用インターフェースモジュール
- `POSIX` – POSIXシステムコマンドへのアクセス用モジュール
- `JSON` – JSONの解析・変換用モジュール(JavaScript Object Notation)
- `Encode::CN` - 中国語エンコード用モジュール
- `Encode::HanExtr` - 中国語エンコードの予備セットモジュール

Webminのバージョンと使用しているWebブラウザによってWebインターフェースのレイアウトが異なることがあります。

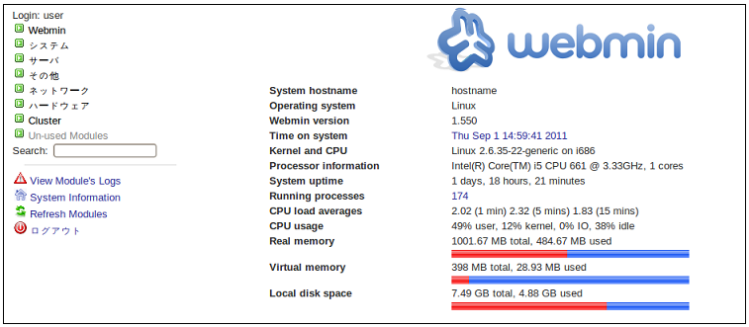
インストール

Dr.Web console for UNIX Internet gatewaysのインストールは以下の手順で行います。

- Webminのインストール
- Webminのプラグインモジュール **Dr.Web console for UNIX Internet gateways**のインストール(`%bin_dir/web/`)

Webminの設定、モジュールのインストールは、WebminのWebインターフェースで行います。

図 17. Webmin メインページ



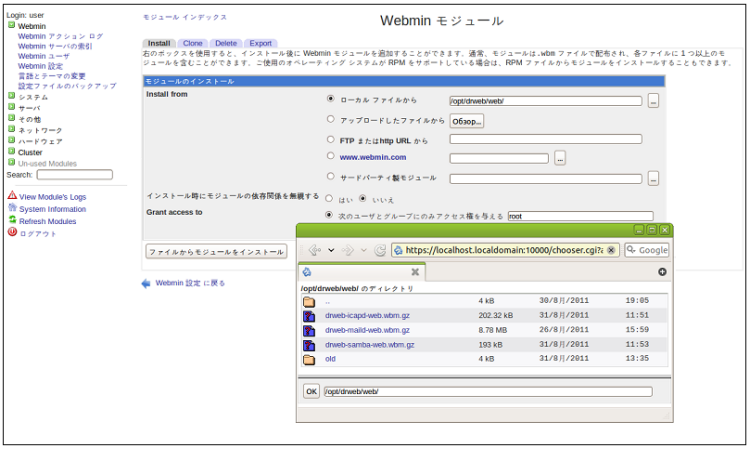
新しいモジュールのインストールは、**Webmin** 設定ページの **Webmin** モジュール で行います。

図 18. Webmin 設定

以下の手順でモジュールのインストールを行います。

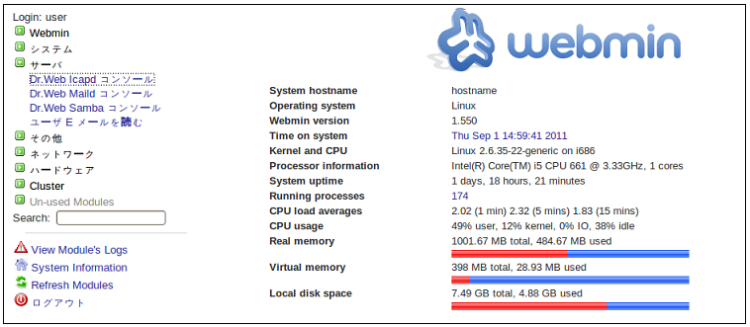
1. **Webmin** モジュールページで、ローカル ファイルからテキストフィールドの右にある **Browse** ボタンを押します。
2. インストールパッケージを選択します。(`%bin_dir/web/`にある**Dr. Web console for UNIX Internet gateways**のパッケージを選択します。)

図 19. Webmin モジュール



3. インストールパッケージを選択後、ファイルからモジュールをインストールボタンを押してインストールを開始します。
4. インストールが成功すると、　　サーバセクションのメニューに**Dr.Web console for UNIX Internet gateways**が表示されます。

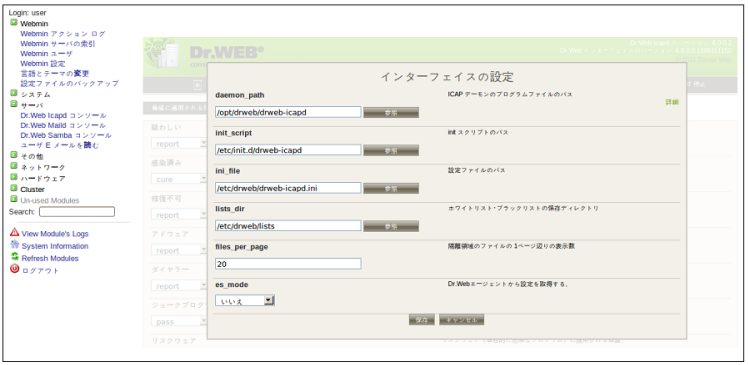
図 20. Dr.Web console for UNIX Internet gateways



基本設定

Dr.Web console for UNIX Internet gatewaysページのインターフェイスの設定 では、設定ファイルのパスやinitスクリプト、ブラックリストおよびホワイトリストを含むディレクトリのパスのほか、**隔離**の1ページあたりの表示件数、デバッグ情報の表示の有無を指定することができます。

図 21. Module configuration

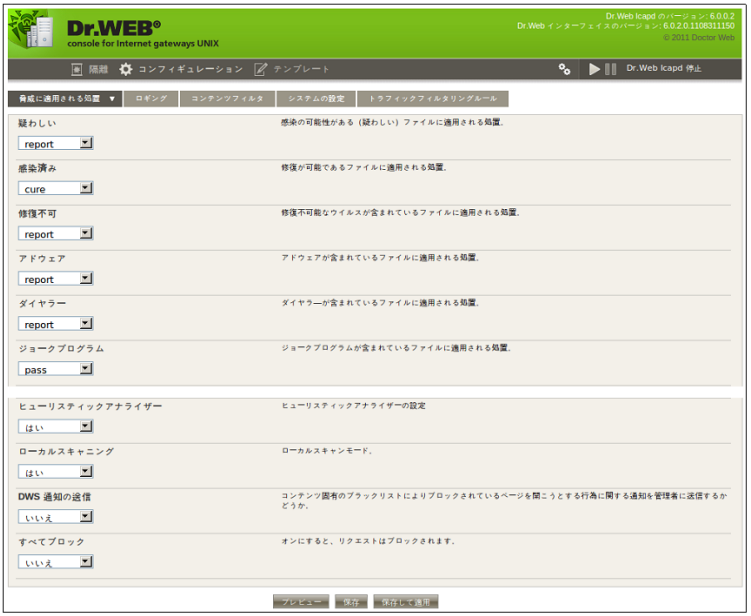


通常、**Quarantine**の1ページあたりの表示件数（Files per page in Quarantine Management）とデバッグ情報の表示（Show debug Information）以外の項目は、変更しないでください。

ユーザーインターフェース

Dr.Web console for UNIX Internet Gatewaysでは、ブラウザの**Back**（戻る）ボタンを使用しないでください。

図 22. Dr.Web console for UNIX Internet gateways



画面右上にDr.Web ICAPとDr.Web for Unix Internet gatewaysWebインターフェースのバージョン情報が表示されています。

バージョン情報の下には、隔離，コンフィギュレーション，テンプレートの3つのセクションがあり、コンフィギュレーションセクションで設定を確認することができます。

セクションの右側では、以下の3つのボタンがあり、現在のステータスを確認することができます。

インターフェイスの設定  Dr.Web Icapdの起動  Dr.Web Icapdの停止 

設定

Configurationセクションは、以下の5つのカテゴリに分かれています。

- Actions Applied to Threats - 検出した脅威に対する処理の設定
- Content Filter - URLフィルタリングの設定
- System settings - ライセンスキーファイルや隔離ディレクトリのパス、管理者の電子メールアドレスの指定などの各種システム設定
- Traffic Filtering Rules - MIMEルールによるフィルタリングルールの設定
- Logging - ログ機能の設定

変更した設定内容を確認する場合は、**Preview** ボタンを押します。プレビューページで、変更前と変更後の設定内容を確認することができます。

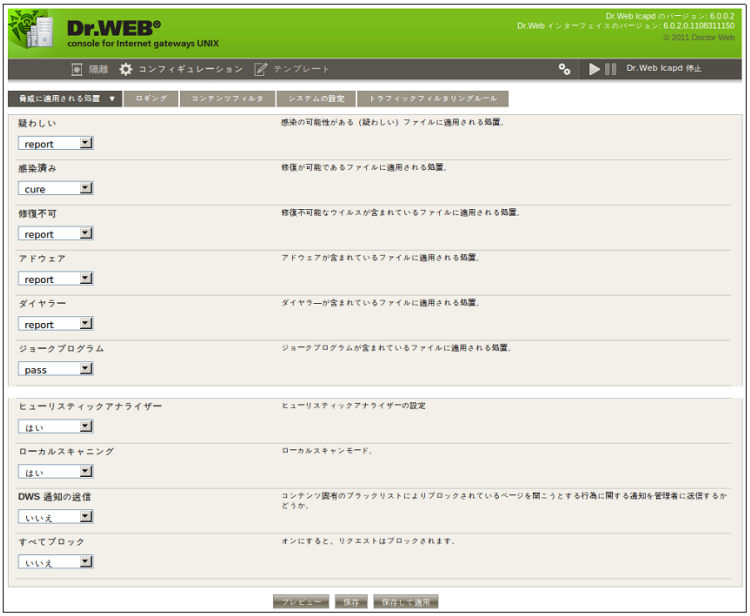
設定を追加する場合は、**Continue Editing** ボタンを押し、設定を取り消す場合は、**Cancel** ボタンを押します。設定を保存する場合は、**Save** または、**Apply and Save** ボタンを押します。

図 23. プレビュー画面

脅威に対するアクション

脅威に対するアクションのページでは、検出した脅威に対して適用する動作などの指定が行えます。ドロップダウンで適用する動作を選択することができます。利用可能な動作の詳細については、**詳細** を参照してください。

図 24. 脅威に対するアクション



ログ

図 25. ログ

パラメータ値の指定は、ドロップダウンによる選択または、対応するテキストフォール度に入力することで行います。

システムイベントを指定したファイルへ出力したい場合は、**Use syslog**のチェックボックスを外し、**Browse** ボタンを押してください。

コンテンツフィルタ

コンテンツフィルタのページでは、ブラックリストを用いたURLフィルタリングの設定が行えます。

図 26. コンテンツフィルタ

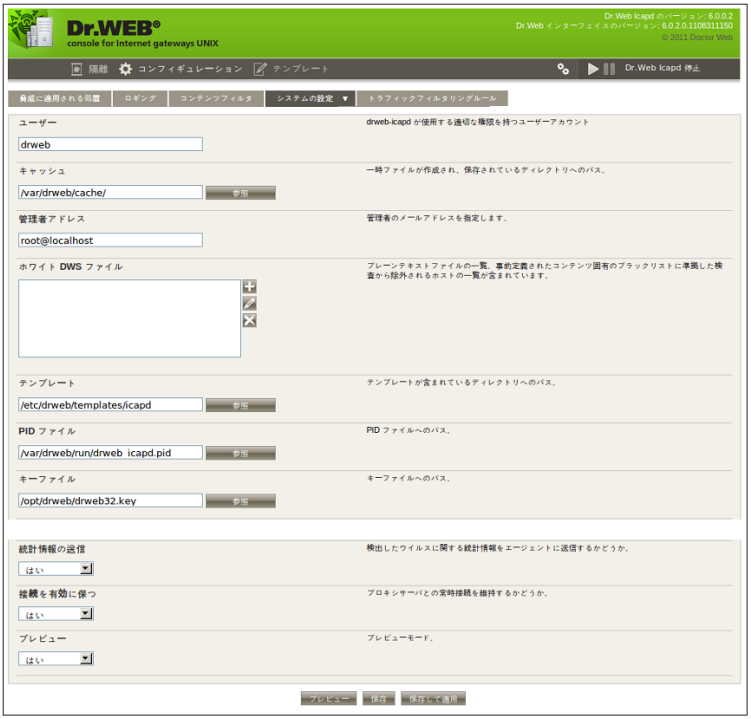
各種ブラックリストによる検査について、**Yes** または **No**を指定することができます。

ブラックリストを手動で定義する場合は、**DwsDirectory**パラメータのテキストフィールドに入力するか、**Browse** ボタンを押して選択してください。

システム設定

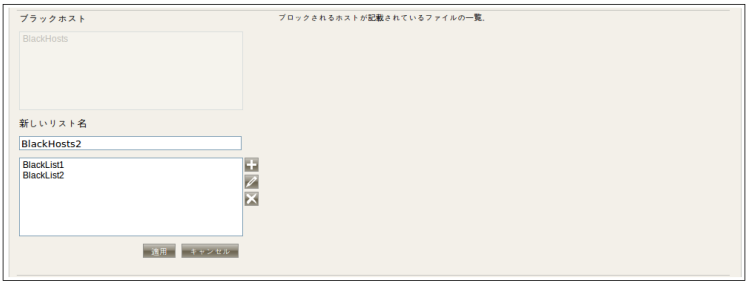
システム設定のページでは、ユーザ定義のブラックリストおよびホワイトリストを設定することができます。また、管理者のメールアドレスの指定やライセンスキーファイル、隔離ディレクトリのパスの指定などが行えます。

図 27. システム設定



BlackHostsセクションで、ユーザ定義のブラックリストを作成することができます。 ボタンを押して、新しいリスト名とホスト名を入力してください。

Figure 28. Creating user-defined list



WhiteHostsおよび**WhiteDwsFiles**セクションで、ユーザ定義のホワイトリストを作成することができます。

WhiteHostsで指定されたホストはウイルス検査が行われません。また、**WhiteDwsFiles**で指定されたホストはURLフィルタリングのブラックリストによる検査が行われません。

トラフィックフィルタリングルール

トラフィックフィルタリングルールのページでは、MIMEルールによるフィルタリングの設定が行えます。メディアタイプは、`TYPE/Format`の形式で、multipart/x-zipやaudio/mpegのように指定します。

図 29. Traffic Filtering Rules

隔離

隔離のページでは、隔離ディレクトリに保存されたコンテンツの一覧が確認できます。タイトルの**URL**, **Size**, **Date**をクリックすることで一覧をソートして表示することができます。

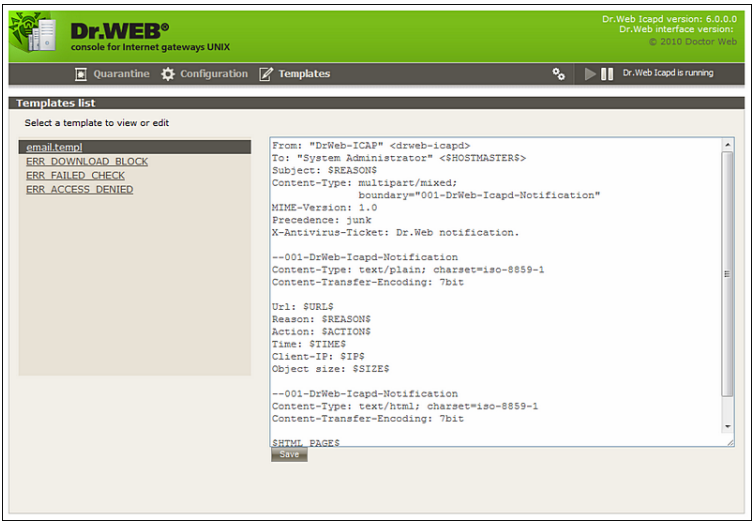
図 30. 隔離

また、左側にあるチェックボックスにチェックを入れて、**Delete** ボタンを押すことで隔離したファイルを削除することができます。

テンプレート

テンプレートのページでは、各種通知、メッセージの編集が行えます。

図 31. テンプレート



`ERR_FAILED_CHECK` – ファイルの検査エラー通知のテンプレート

`ERR_DOWNLOAD_BLOCK` – URLフィルタリングにより、`Web`ページへのアクセスがブロックされたときの通知メッセージのテンプレート

`ERR_ACCESS_DENIED` – ダウンロードファイルが検出されたときの通知メッセージのテンプレート

`email.templ` – 管理者への通知メッセージのテンプレート

お問い合わせ

Dr.Web for Unix Internet gateways は常に改良され続けています。ご利用頂けるアップデートについての最新のニュースおよび情報は、以下のwebサイト上でご覧いただけます。

http://www.drweb.com/

セールス部門:

http://buy.drweb.com/

テクニカルサポート:

http://support.drweb.com/

問題が発生した場合にお送りいただくレポートには次の事柄を記載してください。

- お使いのOSの名称およびバージョン
- **Dr.Web for Unix Internet gateways**モジュールのバージョン
- 全てのモジュールの設定ファイル
- 全てのモジュールのログファイル

付録　ライセンスポリシー

Dr.We® for Unix Internet gateways は、«universal»および«economy» **Dr.Web for Unix Internet gateways**キットの一部、または別々の製品としてご利用いただけます。ライセンスの種類はそれぞれ異なります。

全てのライセンスには期限があります（例：1、2、または3年）。保護するファイルサーバの数もそれぞれ異なります。ライセンス期限、その定量パラメータ、制限は**Doctor Web**のリージョナルパートナーによって異なる場合があり、また今後変更される可能性があります。地域ごとのライセンス条件についての詳細は、お住まいの地域のパートナーまでお問い合わせください。**Doctor Web**の認定パートナー一覧は http://partners.drweb.com/でご覧いただけます。

ライセンスの有効期間中、クライアントは**Dr.Web Global Updating**システムサーバからアップデートをダウンロードし、**Doctor Web**

Internet gatewaysによる保護

Dr.Web for Unix Internet gatewaysは、ICAPプロトコルをサポートするプロキシサーバと連携して動作するゲートウェイ型のアンチウイルスソリューションです。HTTPとFTPトラフィックの検査をサポートします。本書では、SquidおよびSafeSquidによる設定の説明が含まれています。

Dr.Web for Unix Internet gatewaysは、保護するユーザ数に応じたライセンスになっており、最小25ユーザライセンスから提供しています。