Dr.WEB
for UNIX Internet Gateways

管理者マニュアル

Dr.Web for UNIX Internet Gateways
バージョン11.1
管理者マニュアル
2020/10/12

Doctor Webロシア本社

2-12A, 3rd str. Yamskogo polya, Moscow, Russia, 125040

ウェブサイト: https://www.drweb.com/

電話番号：+7 (495) 789-45-87

支社および海外オフィスについては、Doctor Web公式サイトをご覧ください。

Doctor Web

Doctor Web は、悪意のあるソフトウェアやスパムからの効果的な保護を提供するDr.Web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。

Doctor Web のカスタマーは、世界中のホームユーザーから政府機関、小規模な会社、大企業にまで広がっています。

Dr.Web アンチウイルスソリューションは、マルウェア検出における継続的な卓越性と国際情報セキュリティ基準への遵守によって1992年よりその名を広く知られています。

Dr.Web ソリューションに与えられたロシア連邦による認定や数々の賞、そして世界中に広がるそのユーザーが、製品に対する並外れた信頼の何よりの証です。

Dr.Web 製品に対するサポートについて、すべてのカスタマーに対して厚く御礼申し上げます。

目次

表記規則と略語

本マニュアルでは、以下の文字・記号を使用しています。

文字・記号	説明
	重要な部分や指示
	重要な注釈、またはエラーを引き起こす可能性のある状況に関する警告
アンチウイルスネットワーク	新しい用語、または強調したい用語
<IP-address>	プレースホルダー
保存	ボタン、ウィンドウ、メニューアイテム、および他のプログラムインターフェースエレメントの名称
CTRL	キーボードのキー名称
`/home/user`	ファイルやフォルダの名前、コード例
付録 A	本書の他のページや外部Webページへのリンク

本マニュアルでは、(端末または端末エミュレーターで)キーボードから入力されるコマンドラインのコマンドには、コマンドプロンプト記号 `$` または `#` が付いています。この記号は、該当するコマンドの実行に必要な権限を表しています(UNIX系システムの標準的規則に従って)。

`$` - コマンドはユーザー権限で実行できることを示します。

`#` - スーパーユーザー(通常は *root*)権限でコマンドを実行できることを示します。権限を昇格するには、**su**と**sudo**コマンドを使用してください。

略語のリストは、セクション付録G. 略語のリストにあります。

はじめに

Dr.Web for UNIX Internet Gatewaysをお買い上げいただきありがとうございます。本製品は、最先端のウイルス検出および駆除テクノロジーを活用して、さまざまなタイプのコンピューター脅威からサーバーとユーザーを確実に保護します。

このマニュアルの目的は、**GNU/Linux**ファミリーのOSや、**FreeBSD**などの他のUNIX系OSを実行しているサーバーの管理者が、Dr.Web for UNIX Internet Gatewaysバージョン11.1をインストールしてご使用いただけるように支援することです。

ファイルパスの表記法

Dr.Web for UNIX Internet Gatewaysは、さまざまな**UNIX**ベースのOSで動作します。ファイルとコンポーネントへの実際のパスは、OSによって異なります。本書では、次の表記法を使用しています。

- *<opt_dir>* - 主な製品ファイルがあるディレクトリ（実行ファイルとライブラリを含む）。
- *<etc_dir>* - 設定ファイルとキーファイルがあるディレクトリ。
- *<var_dir>* - サポートと製品の一時ファイルが配置されているディレクトリ。

さまざまなOSの規則に対応する実際のパスは、以下の表に示されています。

OSの種類	文字・記号	実際のパス
GNU/Linux	*<opt_dir>*	`/opt/drweb.com`
	<etc_dir>	`/etc/opt/drweb.com`
	<var_dir>	`/var/opt/drweb.com`
FreeBSD	*<opt_dir>*	`/usr/local/libexec/drweb.com`
	<etc_dir>	`/usr/local/etc/drweb.com`
	<var_dir>	`/var/drweb.com`

スペースを考慮して、例では**GNU/Linux** OSのパスを使用しています。本書では、可能な場合においてすべてのOSの実際のパスが例に使用されています。

この製品について

このセクションには、Dr.Web for UNIX Internet Gatewaysに関する次の情報が含まれています。

機能

Dr.Web for UNIX Internet Gatewaysは、UNIX(**GNU/Linux**、**FreeBSD**)で実行されているインターネットゲートウェイやプロキシを、ウイルスや他のタイプの悪意のあるソフトウェアから保護し、さまざまなプラットフォーム向けに作成された脅威の拡散を防ぐためのものです。

主なコンポーネント(スキャンエンジンとウイルスデータベース)は、非常に効果的でリソースを節約するだけでなく、クロスプラットフォームでもあるため、Doctor Webスペシャリストはさまざまなプラットフォームをターゲットとした脅威から一般的なOSのコンピューターやモバイルデバイスを守る、信頼性の高いアンチウイルスソリューションを作成できます。現在、Doctor WebではDr.Web for UNIX Internet Gatewaysとともに、**UNIX**ベースのOS(**GNU/Linux**、**FreeBSD**)と**IBM OS/2**、**Novell NetWare**、**macOS**と**Windows**の両方のアンチウイルスソリューションを提供しています。さらに、**Android**、**Symbian**、**BlackBerry**、Windows Mobileを実行するデバイスを保護するための、他のアンチウイルス製品が開発されています。

Dr.Web for UNIX Internet Gatewaysのコンポーネントは常に更新され、ウイルスデータベース、Webリソースカテゴリーのデータベース、メールメッセージのスパムフィルタリングのルールデータベースには定期的に新しいシグネチャが追加されるため、サーバー、ワークステーション、モバイルユーザーとそのプログラムやデータに最新の保護を提供します。未知のウイルスに対する追加の保護を提供するため、スキャンエンジンとDr.Web Cloudサービスにはヒューリスティック解析が実装されており、シグネチャがデータベースにない最新の脅威に関する情報を保存します(この機能は一部の製品でのみ利用できます)。

メイン機能

Dr.Web for UNIX Internet Gateways のメイン機能

1. **脅威の検出と駆除。**悪意のあるプログラム(メールファイルやブートレコードに感染するものを含むウイルス、トロイの木馬、メールワームなど)や不要なソフトウェア(アドウェア、ジョークプログラム、ダイアラーなど)を検索します。コンピューターの脅威の種類に関する詳細については、付録 A. コンピューター脅威の種類を参照してください。

   脅威の検出方法:

   - *シグネチャ解析*。既知の脅威の検出を可能にします。
   - *ヒューリスティック解析*。ウイルスデータベースに含まれていない脅威の検出を可能にします。
   - *クラウドベースの脅威検出テクノロジー*。Dr.Web Cloudサービスを使用して、最近の脅威に関する最新情報を収集し、Dr.Web製品に送信します。

ヒューリスティックアナライザは誤検知を引き起こす可能性があることに注意してください。したがって、アナライザによって検出された脅威を含むオブジェクトは「疑わしい」と見なされます。このようなファイルを隔離し、解析のためにDoctor Webアンチウイルスラボに送信することをお勧めします。脅威を駆除する方法の詳細は、付録 B. コンピューター脅威の駆除を参照してください。

ユーザーの要求に応じてファイルシステムをスキャンする場合、ユーザーが利用できるすべてのファイルシステムオブジェクトのフルスキャン、または指定されたオブジェクトのみ（指定された基準を満たす個別のディレクトリまたはファイル）のカスタムスキャンが可能です。さらに、システム内で現在アクティブなプロセスをサポートするボリュームと実行ファイルのブートレコードを別々にチェックすることもできます。後者の場合、脅威が検出されると、悪意のある実行ファイルを駆除するだけでなく、選択的スキャンにより実行されているすべてのプロセスが強制的に終了されます。一連の異なるアクセスレベルを持つファイルへのアクセスの必須モデルを実装するシステムでは、現在のアクセスレベルでは利用できないファイルのスキャンは特別な自律コピーモードで行うことができます。

ファイルシステムで検出された脅威を含むすべてのオブジェクトは、自律コピーモードで検出された脅威を除いて、永久保存された脅威レジストリに登録されます。

Dr.Web for UNIX Internet Gatewaysに含まれているDr.Web Ctlコマンドラインを使うと、SSHまたはTelnet経由でのリモート端末アクセスを提供するリモートネットワークホストの脅威ファイルシステムをスキャンできます。

> リモートスキャンは、リモートホスト上の悪意ファイルや疑わしいファイルの検出にのみ使用できます。リモートホストで検出された脅威を排除するには、このホストによって直接提供される管理ツールを使用する必要があります。たとえば、ルーターやその他の「スマート」デバイスの場合は、ファームウェア更新のメカニズムを使用できます。コンピューティングマシンの場合は、コンピューティングマシンへの接続（任意でリモート端末モードを使用）とファイルシステムのそれぞれの操作（ファイルの削除または移動など）、またはコンピューティングマシンにインストールされているアンチウイルスソフトウェアの実行を介して実行できます。

2. **インターネットに送信されたデータの分析**。ユーザーの要求（つまり、Webサーバーに接続してファイルを送信しようとする試み）だけでなく、ユーザーの要求に応じて送信されるデータも監視されます。要求や送信されるデータを分析するために、Dr.Web for UNIX Internet GatewaysはICAPプロトコル経由でプロキシサーバーへの外部フィルターとして接続し、ローカルネットワークユーザーのHTTP接続を処理します。さらに、SpIDer Gateコンポーネントを使用すると、組織の公開サーバーによる感染したファイルの送受信を防ぐバリア機能を実行できます（*このオプションは***GNU/Linux***にのみ使用可能です*）。不要なWebサイトへのアクセスを制限するために、この製品は自動的に更新されるWebリソースカテゴリーのデータベースを使用します。これらは、システム管理者が手動で作成したホワイトリストとブラックリストであるDr.Web for UNIX Internet Gatewaysとともに提供されます。この製品はDr.Web Cloudサービスも参照して、インターネットリソースが他のDr.Web製品によって悪意のあるものとしてマークされているかどうか情報を確認します。
3. **感染したオブジェクトや疑わしいオブジェクトを確実に隔離します**。サーバーのファイルシステムで検出されたそのようなオブジェクトは、システムへの害を防ぐ特別なフォルダに移動され、隔離されます。隔離へ移動されたオブジェクトは、特別なルールに従って名前が変更され、必要に応じて、オンデマンドでのみ元の場所に復元できます。

   HTTPプロトコルメッセージ内のDr.Web ICAPDコンポーネントによって検出された脅威は、インターネットゲートウェイ上の検疫に移動されません。その代わりに、それらのロードと受信者への転送はブロックされ、ユーザーにはブロックに関するメッセージを含む特別なHTMLページが通知されます。
4. マルウェアに対する高度な保護を維持するための、スキャンエンジン、ウイルスデータベース、Webリソースカテゴリーのデータベースの**自動更新**。
5. ウイルスイベントに関する**統計の収集**、脅威検出イベントのロギング。SNMPを介して検出された脅威に関する通知を外部のモニタリングシステムと集中管理サーバーに（Dr.Web for UNIX Internet Gatewaysが集中管理モードで動作している場合はDr.Web Cloudにも）送信。

6. **集中管理モードでの動作**（Dr.Web Enterprise Serverなどの集中管理サーバーに接続されている場合、またはDr.Web AV-Deskサービスの一部として）。このモードによって、保護されたネットワーク内のコンピューターに統一されたセキュリティポリシーを実装することができます。企業のネットワーク、プライベートネットワーク（VPN）、またはサービスプロバイダーのネットワーク（インターネットサービスプロバイダーなど）のいずれかです。

Dr.Web for UNIX Internet Gatewaysの構成

Dr.Web for UNIX Internet Gatewaysは複数のコンポーネントで構成され、各コンポーネントには独自の機能のセットがあります。コンポーネントの目的に応じて、次のカテゴリーに分類されます。

- 基本アンチウイルスコンポーネント：Dr.Web for UNIX Internet Gatewaysコアを形成。このカテゴリーのコンポーネントがない場合、製品はファイル（およびその他のデータ）をスキャンできず、ウイルスやその他の脅威を検出できません。
- 脅威検索コンポーネント。これらのコンポーネントは、Dr.Web for UNIX Internet Gatewaysの基本的なタスクを解決するために使用されます。これらのコンポーネントは、動作時に基本アンチウイルスコンポーネントを使用します。
- サービスコンポーネント：補助的アンチウイルス保護の問題を解決（アンチウイルスデータベースの更新、集中管理サーバー接続、一般的なDr.Web for UNIX Internet Gateways動作管理など）。
- インターフェースコンポーネント：（ユーザーまたはサードパーティのアプリケーション）にDr.Web for UNIX Internet Gatewaysのインターフェースを提供。

以下は、Dr.Web for UNIX Internet Gatewaysコンポーネントのリストです。

1.基本アンチウイルスコンポーネント

コンポーネント	説明
Dr.Web Virus-Finding Engine	スキャンエンジン。ウイルスと悪意のあるプログラムを検出するアルゴリズムを実装します（シグネチャ解析とヒューリスティック解析を使用）。 スキャンエンジンはDr.Web Scanning Engine管理によって動作しています。 実行ファイル：**drweb32.dll** ログファイルに出力される内部名：`CoreEngine`
Dr.Web Scanning Engine	スキャンエンジン。このコンポーネントは、Dr.Web Virus-Finding Engineスキャンエンジンとアンチウイルスデータベースをロードします。 • スキャンのためにファイルの内容とブートレコードをスキャンエンジンに送信します。 • スキャンを待機しているファイルをキューに入れます。 • このアクションが適用可能な脅威を修復します。 Dr.Web ConfigDによって動作するか、自律的に動作できます。 Dr.Web File CheckerおよびDr.Web Network Checkerコンポーネントで使用されます。また、Dr.Web MeshDコンポーネント（特定のモード）、およびDr.Web Scanning Engine APIで使用される外部（Dr.Web for UNIX Internet Gatewaysに対して）アプリケーションで使用できます。

コンポーネント	説明
	実行ファイル：**drweb-se** ログに表示される内部名：ScanEngine
ウイルスデータベース	ウイルスデータベース。ウイルスのシグネチャやその他の脅威のデータベースと悪質なソフトウェアの検出と駆除のアルゴリズムは自動で更新されます。 スキャンエンジンDr.Web Virus-Finding Engineで使用され、一緒に提供されます。
Webリソースカテゴリーのデータベース	Web分類されたリソースのリストを含む、自動的に更新されるデータベース。不要なWebサイトを識別するために使用されます。 SpIDer Gate、Dr.Web ICAPD、Dr.Web MailDなど、ユーザーとアプリケーションのネットワークアクティビティをスキャンするコンポーネントによって使用されます。
Dr.Web File Checker	ファイルシステムコンポーネントと隔離マネージャーをスキャンするためのコンポーネント。 • ローカル（Dr.Web Scanning Engineに対して）ファイルシステムのファイルをスキャンするときに、脅威スキャンコンポーネントからタスクを受け取ります。 • タスクに従ってファイルシステムディレクトリをサーフし、スキャンするファイルをDr.Web Scanning Engineエンジンに送信し、スキャンの進行状況をクライアントコンポーネントに通知します。 • 感染したファイルの削除、隔離への移動、隔離からの復元、隔離ディレクトリの管理を行います。 • キャッシュを構築し、最新の状態に保ちます。キャッシュには、以前にスキャンされたファイルに関する情報が含まれており、ファイルを再スキャンする周期を減らします。 ファイルシステムオブジェクトをスキャンするコンポーネントによって使用されます。 実行ファイル：**drweb-filecheck** ログに表示される内部名：FileCheck
Dr.Web Network Checker	ネットワークデータスキャンエージェント。 • 実際にスキャンを実行する際、スキャンエンジンにデータを送信するのに使用されます。データは、ネットワーク経由で製品のコンポーネントによって送信されます（Dr.Web ClamD、SpIDer Gate、Dr.Web ICAPDなどのコンポーネント）。 • Dr.Web for UNIX Internet Gatewaysが、リモートホストからまたはリモートホストへのスキャンにファイルを送受信するファイルの分散スキャンを手配することが可能になります。そのためには、リモートホストにはUNIXオペレーティングシステム用にインストールされ実行されているDr.Webを備えている必要があります。分散スキャンモードでは、多数のスキャンタスクのあるホスト（メールサーバー、ファイルサーバー、インターネットゲートウェイなど）の負荷を減らすことで、リモートホスト間でスキャン負荷を自動的に分散できます。

コンポーネント	説明
	スキャン用のデータを受信できるネットワーク内のパートナーホストが存在する場合、スキャンにDr.Web Network Checkerを使用するコンポーネントはローカルのDr.Web Scanning Engineエンジンを使用しない場合があります。したがって、Dr.Web Scanning Engine、Dr.Web Virus-Finding Engine、およびアンチウイルスデータベースがない可能性があります。 セキュリティ上の理由から、ファイルはSSLで転送されます。 実行ファイル：**drweb-netcheck** ログに表示される内部名：`NetCheck`
Dr.Web MeshD	Dr.Web for UNIX Internet Gatewaysをローカルクラウドに接続するコンポーネント。Dr.Web for UNIX製品が更新およびファイルスキャンの結果を交換し、スキャンのためにファイルを相互に送信し、スキャンエンジンサービスを直接提供できるようにします。 *このコンポーネントが製品に含まれている他、コンポーネントが接続されているローカルクラウドとスキャンエンジンサービスを提供するホストが存在する場合、Dr.Web Scanning Engine、Dr.Web Virus-Finding Engine、およびアンチウイルスデータベースがない可能性があります。* 実行ファイル：**drweb-meshd** ログに表示される内部名：`MeshD`

2.脅威検索コンポーネント

コンポーネント	説明
Dr.Web ICAPD	ICAPをサポートするHTTPプロキシ（**Squid**など）を通過するリクエストとネットワークトラフィックを分析するICAPサーバー。 また、感染したファイルの送信や、システム管理者が作成したインターネットリソースカテゴリーやブラックリストに属するネットワークホストへのアクセスを防止します。外部サーバーへのアクセスが禁止されており、送信されたデータに脅威が含まれている場合は、要求されたリソースにアクセスできない、または送信されたファイルが感染していることを知らせる特別なページをユーザーに返します。 Dr.Web Network Checkerを使用して、プロキシサーバーから受信したデータをスキャンします。 実行ファイル：**drweb-icapd** ログに表示される内部名：`ICAPD`
SpIDer Gate	ネットワークトラフィックとURLを監視するためのコンポーネント。 ネットワークからローカルホストにダウンロードされ、そこから外部ネットワークに送信されたデータの脅威をスキャンするように設計されています。また、コンポーネントは、Webリソースの不要なカテゴリーだけでなく、システム管理者が作成したブラックリストにも含まれるネットワークホストとの接続を阻止します。

コンポーネント	説明
	Dr.Web Network Checkerを使用して、受信したデータをスキャンします。  **GNU/Linux**OS用ディストリビューションにのみ含まれています。 実行ファイル：**drweb-gated** ログに表示される内部名：GateD
Dr.Web Firewall for Linux	ネットワーク接続モニター。 SpIDer Gateによって使用され、送信されたネットワークトラフィックをスキャンするためにサーバー上で動作するアプリケーションに接続ルーティングを提供します。  **GNU/Linux**OS用ディストリビューションにのみ含まれています。 実行ファイル：**drweb-firewall** ログファイルに出力される内部名：LinuxFirewall

3.サービスコンポーネント

コンポーネント	説明
Dr.Web ConfigD	Dr.Web for UNIX Internet Gateways設定デーモン。次の機能を実行します。 • 設定に応じて、製品のコンポーネントを起動／停止します。 • 動作に障害が発生した場合、コンポーネントを自動的に再起動します。他のコンポーネントのリクエストに応じてコンポーネントを起動します。別のコンポーネントが起動またはシャットダウンしたときにアクティブなコンポーネントに通知します。 • 現在のライセンスキーと設定に関する情報を保存し、このデータをすべてのコンポーネントに提供します。それらの情報を提供すると予測されるDr.Web for UNIX Internet Gatewaysのコンポーネントから設定とライセンスキーを受け取ります。ライセンスキーと設定の変更について他のコンポーネントに通知します。 実行ファイル：**drweb-configd** ログファイルに出力される内部名：ConfigD
Dr.Web ES Agent	集中管理エージェント集中管理モードとモバイルモードでの製品動作の組み合わせを確実なものにします。 • 製品と集中管理サーバー間の接続、ライセンスキーファイル、ウイルスデータベースとコンポーネントの更新。

コンポーネント	説明
	• Dr.Web for UNIX Internet Gatewaysに含まれるコンポーネントとその状態、ウイルスイベントの統計に関する情報をサーバーに送信します。 実行ファイル：**drweb-esagent** ログに表示される内部名：ESAgent
Dr.Web Updater	更新コンポーネント ウイルスデータベース、Webリソースカテゴリーのデータベース、スキャンエンジン。 アップデートはスケジュールに従い、またユーザーの要求に応じて（Dr.Web Ctlまたは管理Webインターフェース経由で）自動的にダウンロードできます。 実行ファイル：**drweb-update** ログに表示される内部名：Update
Dr.Web CloudD	Dr.Web Cloudインタラクションコンポーネント。 Dr.Web Cloudサービスに、閲覧したURLとスキャンしたファイルに関する情報を送信するコンポーネントです。これらの情報は、ウイルスデータベースにまだ記載されていない脅威をチェックします。 実行ファイル：**drweb-cloudd** ログに表示される内部名：CloudD
Dr.Web LookupD	外部データソースからデータを取得するためのコンポーネント。 外部データソース（ディレクトリサービス、ファイル、相対データベースなど）からデータを取得します。データはトラフィックモニタリングのルールで使用されます。 実行ファイル：**drweb-lookupd** ログに表示される内部名：LookupD
Dr.Web StatD	Dr.Web for UNIX Internet Gatewaysコンポーネントの動作イベントを格納するためのコンポーネント。 プログラムの複雑なイベント（異常終了、脅威の検出など）のストレージを受け取り、管理します。 実行ファイル：**drweb-statd** ログファイルに出力される内部名：StatD

4.インターフェースコンポーネント

コンポーネント	説明
Dr.Web HTTPD	Dr.Web for UNIX Internet Gateways管理Webサーバー。 Dr.Web for UNIX Internet Gatewaysコンポーネントを管理するためのカスタムHTTP APIを提供します。

コンポーネント	説明
	指定されたAPIは、管理Webインターフェース（個別にインストールする必要があります）。 セキュリティ上の理由から、WebインターフェースはHTTPS経由でユーザーと対話します。 Dr.Web Network Checkerを使用して、スキャン用のデータをDr.Web Scanning Engineに送信します。 実行ファイル：**drweb-httpd** ログに表示される内部名：HTTPD
Dr.Web Web管理インターフェース	管理Webインターフェース ローカルホストまたはリモートホスト上の任意のブラウザからインターフェースにアクセスできます。Webインターフェースにより、製品はサードパーティのWebサーバー（**Apache HTTP Server**など）も**Webmin**などのリモート管理ツールも使用できません。 コンポーネントの機能は、Dr.Web HTTPDコンポーネントによって提供されます。
Dr.Web Ctl	コマンドラインからDr.Web for UNIX Internet Gatewaysを管理するためのツール。 ユーザーがファイルスキャンを開始し、隔離されたオブジェクトを表示し、ウイルスデータベースの更新手順を開始し、Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続または切断し、パラメータを表示、設定できるようにします。 実行ファイル：**drweb-ctl** ログに表示される内部名：Ctl
Dr.Web SNMPD	SNMPエージェント。 Dr.Web for UNIX Internet GatewaysとSNMP経由の外部監視システムとの統合用に設計されています。このような統合により、製品のコンポーネントの状態を監視したり、脅威の検出と駆除に関する統計を収集したりできます。 SNMP v2cとv3をサポートします。 実行ファイル：**drweb-snmpd** ログに表示される内部名：SNMPD
Dr.Web ClamD	アンチウイルスデーモンである**clamd**（**ClamAV®**アンチウイルスのコンポーネント）のインターフェースをエミュレートするコンポーネント。 **ClamAV®**をサポートするすべてのアプリケーションが、アンチウイルススキャンにDr.Web for UNIX Internet Gatewaysを透過的に使用できるようになります。 モードに応じて、Dr.Web Network CheckerまたはDr.Web File Checkerを使用して、スキャン用のデータをDr.Web Scanning Engineに送信します。 実行ファイル：**drweb-clamd** ログに表示される内部名：ClamD

下の図は、Dr.Web for UNIX Internet Gatewaysの構造とその外部アプリケーションとの動作を示しています。

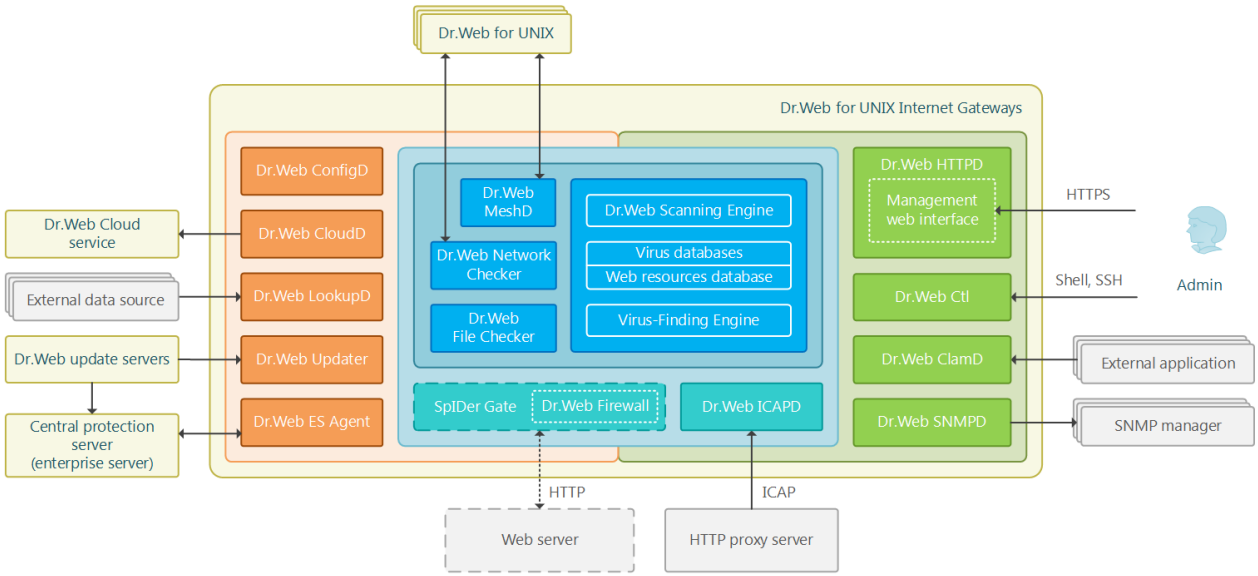

図1. Dr.Web for UNIX Internet Gatewaysの構造

このスキームでは、次の表記法が使用されています。

	- Dr.Web for UNIX Internet Gatewaysのコンポーネントおよびソリューションに含まれないDr.Webのシステムや Dr.Webのアプリケーション。
	- Dr.Web for UNIX Mail Serversと連携する外部プログラムや製品。
	- 特定のアンチウイルス保護タスク(アンチウイルスデータベースの更新、集中管理サーバーへの接続、一般的な動作管理など)を実行するサービスコンポーネント。
	- (ユーザーまたはサードパーティ製アプリケーションに) Dr.Web for UNIX Internet Gatewaysのインターフェースを提供するコンポーネント。
	- アンチウイルススキャンに使用されるコンポーネント。
	- Dr.Web for UNIX Internet Gatewaysコアを形成する基本的なアンチウイルスコンポーネント。データとファイルのスキャンを実行するコンポーネントによって使用されます。

Dr.Web for UNIX Internet Gatewaysのディストリビューションと使用状況によっては、破線でマークされたコンポーネントが含まれない場合があります。

Dr.Web for UNIX Internet Gatewaysコンポーネントの詳細については、Dr.Web for UNIX Internet Gatewaysコンポーネントを参照してください。

隔離に移動する

Dr.Web for UNIX Internet Gateways 11.1の隔離ディレクトリは、システムセキュリティに脅威を与え、現在修復できないファイルを分離するのに役立ちます。このような脅威は、Dr.Web for UNIX Internet Gatewaysにとって未知のもの(つまり、ヒューリスティックアナライザによってウイルスが検出されたが、ウイルスのシ

グネチャおよび修復方法がデータベースにないもの）または修復中にエラーを引き起こしたものになります。さらに、ユーザーが検出された脅威のリストでアクションとして隔離を選択した場合や、脅威のタイプごとにアクションとして隔離を設定で指定した場合は、ユーザーの要求に応じてファイルを隔離できます。

隔離されたファイルの名前は特別なルールに従って変更されます。隔離されたファイルの名前を変更することで、ユーザーやアプリケーションによって特定されることを防ぎ、Dr.Web for UNIX Internet Gateways に備わった隔離管理ツールを回避してそれらにアクセスしようとする試みを困難にします。また、ファイルが隔離に移されると、それらを起動させる試みを防ぐために実行ビットがリセットされます。

隔離ディレクトリは以下の場所にあります。

- *ユーザーのホームディレクトリ*（コンピューター上に複数のユーザーアカウントが存在する場合、各ユーザーに個別の隔離ディレクトリが作成される可能性があります）
- ファイルシステムにマウントされた*各論理ボリュームのルートディレクトリ*

Dr.Web隔離ディレクトリの名前は常に`.com.drweb.quarantine`となり、「隔離」アクションが適用されるまで作成されません。その際、オブジェクトを隔離するために必要なディレクトリのみが作成されます。ディレクトリを選択する際は、ファイル所有者の名前を使用します。検索は悪意のあるオブジェクトのある場所から上の階層に向かって行われ、所有者のホームディレクトリに到達した場合、このディレクトリに作成された隔離フォルダが選択されます。そうでない場合、ファイルはボリュームのルートディレクトリ内に作成された隔離内に移されます（これはファイルシステムのルートディレクトリと同じではない場合があります）。したがって、隔離に移動された感染ファイルは常にボリューム上に配置されるため、複数のリムーバブルデータストレージやその他のボリュームがシステム内の異なる場所にマウントされた場合でも、隔離は正しく動作します。

ユーザーは、ユーティリティDr.Web Ctlを使用してコマンドラインから隔離コンテンツを管理できる他、管理用Webインターフェース（インストールされている場合）上からも管理できます。すべてのアクションは隔離全体に適用されます。つまり、変更はその時点で利用可能なすべての隔離ディレクトリに影響します。

隔離されたオブジェクトに対する操作は 有効なライセンス が見つからない場合でも行うことができます。ただし、この場合、隔離されたオブジェクトを修復することはできません。

Dr.Web for UNIX Internet Gatewaysのすべてのアンチウイルスコンポーネントが脅威の分離に隔離を使用できるわけではありません。たとえば、Dr.Web ClamD、Dr.Web ICAPD、Dr.Web MailDのコンポーネントでは使用されません（製品に含まれていない場合があります）。

ファイルのパーミッションと権限

ファイルシステムのオブジェクトをスキャンし、脅威を駆除するために、Dr.Web for UNIX Internet Gateways（を動作させるユーザー）は以下のパーミッションを必要とします。

アクション	必要な権限
検出されたすべての脅威を一覧にする	制限されていません。特別なパーミッションは必要ありません。
アーカイブのコンテンツを一覧表示する	制限されていません。特別なパーミッションは必要ありません。

アクション	必要な権限
（破損した、または悪意のあるエレメントのみを表示する）	
隔離へ移動する	制限されていません。その読み込みまたは書き込み権限に関係なく、ユーザーは感染したすべてのファイルを隔離できます。
脅威を削除する	ユーザーは削除するファイルに対する書き込み権限を持っている必要があります。 コンテナ（アーカイブ、メール添付ファイルなど）内のファイルで脅威が検出された場合は、削除アクションの代わりにコンテナの隔離への移動が実行されます。
修復する	制限されていません。アクセス権限と修復されたファイルの所有者は修復後も変わりません。 検出された脅威を削除することによって修復が可能である場合、ファイルを削除できます。
隔離からファイルを復元する	ユーザーはファイルの読み込み権限と復元先ディレクトリへの書き込み権限を持っている必要があります。
隔離からファイルを削除する	ユーザーは隔離されたファイルへの書き込み権限を持っている必要があります。

スーパーユーザー（*root*）権限でコマンドライン管理Dr.Web Ctlツールの動作を有効にするには、**su**コマンドを使用してユーザーを変更するか、**sudo**コマンドを使用して、他のユーザーとしてコマンドを実行できます。

Dr.Web Scanning Engineスキャンエンジンは、4 GBを超えるサイズのファイルをスキャンできません（このようなファイルをスキャンしようとすると、エラーメッセージ「*ファイルサイズが大きすぎます*」が表示されます）。

動作モード

Dr.Web for UNIX Internet Gatewaysは、スタンドアロンモードでも、*集中管理サーバー*によって管理される*アンチウイルスネットワーク*の一部としても動作できます。*集中管理モード*での操作には、追加のソフトウェアのインストールやDr.Web for UNIX Internet Gatewaysの再インストールまたは削除は必要ありません。

- *スタンドアロンモード*では、保護するコンピューターはアンチウイルスネットワークに接続されず、その操作はローカルで管理されます。このモードでは、設定ファイルとライセンスキーファイルはローカルディスクにあり、Dr.Web for UNIX Internet Gatewaysは保護するコンピューターから完全に制御されます。ウイルスデータベースの更新はDoctor Web更新サーバーから受信します。
- *集中管理モード（エンタープライズモード）*では、コンピューターの保護は集中管理サーバーによって管理されます。このモードでは、Dr.Web for UNIX Internet Gatewaysの一部の機能と設定は、アンチウイルスネットワークに実装されている一般的な（企業の）アンチウイルス保護ポリシーに従って調整できます。集中管理モ

ードでの動作に使用されるライセンスキーファイルは、集中管理サーバーから受信されます。ローカルコンピューターに保存されているデモキーファイルがあっても、そのファイルは使用されません。Dr.Web for UNIX Internet Gatewaysの操作に関する情報とともにウイルスイベントの統計が集中管理サーバーに送信されます。ウイルスデータベースへの更新も集中管理サーバーから受信されます。

- *モバイルモード*では、Dr.Web for UNIX Internet GatewaysはDoctor Web更新サーバーから更新を受信しますが、製品の操作はローカル設定と集中管理サーバーから受信したライセンスキーファイルで管理されます。モバイルモードに切り替えることができるのは、集中管理サーバー設定で許可されている場合のみです。

集中管理のコンセプト

Doctor Webの集中管理ソリューションはクライアント-サーバーモデルを使用します（下図参照）。

ワークステーションとサーバーは*ローカルにインストールされたアンチウイルスコンポーネント*（以下「Dr.Web for UNIX Internet Gateways」）によって脅威から保護されます。これらコンポーネントはリモートコンピューターのアンチウイルス保護を提供し、ワークステーションと集中管理サーバーとの接続を可能にします。

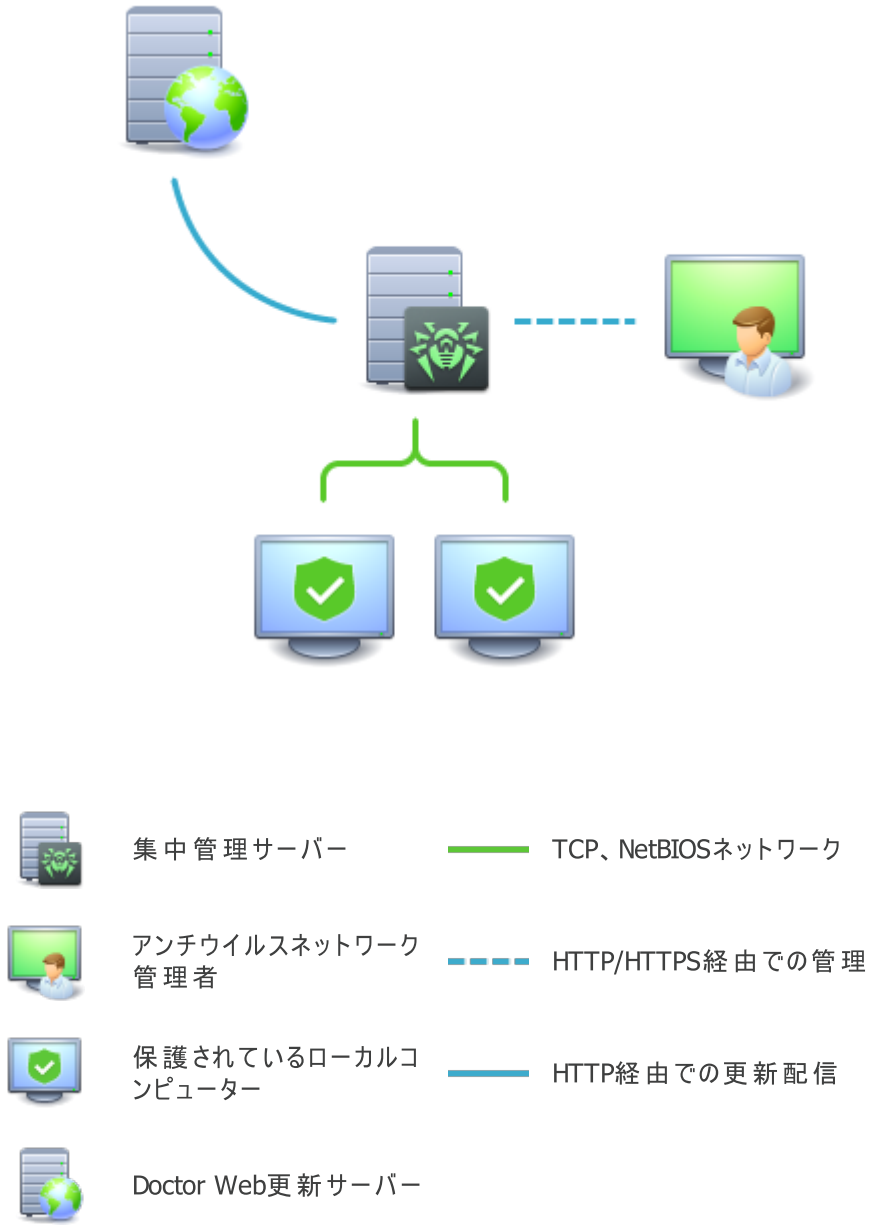

図2. アンチウイルスネットワークの理論的構造

ローカルコンピューターの更新と設定は *集中管理サーバー* から行われます。アンチウイルスネットワーク内の一連の指示やデータ、統計も集中管理サーバーを経由します。保護するコンピューターと集中管理サーバー間のトラフィック量はかなり大きくなる場合があります。そこで、トラフィックを圧縮するオプションを提供しています。機密データの漏洩やコンピューター上にダウンロードされたソフトウェアの置き換えを防ぐため、暗号化もサポートされています。

必要なすべての更新がDoctor Web更新サーバーから集中管理サーバーにダウンロードされます。

ローカルアンチウイルスコンポーネントは、アンチウイルスネットワーク管理者から受け取ったコマンドに従って、集中管理サーバーから設定および管理されます。管理者は集中管理サーバーとアンチウイルスネットワークのトポロジを管理し（たとえば、リモートコンピューターから集中管理サーバーへの接続を検証する）、必要に応じてローカルアンチウイルスコンポーネントの動作を設定します。

ローカルアンチウイルスコンポーネントは、第三者製のアンチウイルス製品、または集中管理モードでの動作をサポートしていない他のDr.Webアンチウイルスソリューション（Dr.Web Anti-virus バージョン5.0など）と互換性がありません。同一コンピューター上に2つのアンチウイルスプログラムをインストールすると、システムクラッシュや重要なデータの紛失を引き起こす場合があります。

集中管理モードでは、集中管理センターを使用して操作レポートをエクスポートおよび保存できます。レポートは、HTML、CSV、PDF、XML形式でエクスポートおよび保存できます。

集中管理サーバーとの接続

Dr.Web for UNIX Internet Gatewaysは、Dr.Web Ctlコマンドラインベース管理ツールの`esconnect`コマンドを使用して、アンチウイルスネットワークの集中管理サーバーに接続できます。

集中管理サーバーを検証するには、サーバーが使用する一意のパブリック暗号化キーに対応する証明書を使用します。デフォルトでは、Dr.Web ES Agent集中管理エージェントは、接続しているサーバーの証明書ファイルを指定しない限り、サーバーへの接続を許可しません。証明書ファイルは、最初に、Dr.Web for UNIX Internet Gatewaysを接続するサーバーが提供するアンチウイルスネットワークの管理者から取得する必要があります。

Dr.Web for UNIX Internet Gatewaysが集中管理サーバーに接続されている場合は、製品をモバイルモードに切り替えたり、集中管理モードに戻したりできます。モバイルモードのオン/オフの切り替えは、Dr.Web ES Agentコンポーネントの**MobileMode**設定パラメータのヘルプを使用して行います。

Dr.Web for UNIX Internet Gatewaysは、集中管理サーバーの設定で許可されている場合にのみ、モバイルモードに切り替えることができます

製品をアンチウイルスネットワークから切断する

Dr.Web for UNIX Internet Gatewaysは、Dr.Web Ctlコマンドラインベース管理ツールの`esdisconnect`コマンドを使用して、アンチウイルスネットワークの集中管理サーバーから切断できます。

システム要件と互換性

このセクションの内容：

システム要件

Dr.Web for UNIX Internet Gatewaysは、以下の要件を満たすコンピューターで使用できます。

コンポーネント	要件
プラットフォーム	以下のアーキテクチャおよびコマンドシステムのプロセッサーがサポートされています。 • **Intel/AMD**: 32ビット（*IA-32*、*x86*）、64ビット（*x86_64*、*x64*、*amd64*） • **ARM64**
ランダムアクセスメモリ（RAM）	少なくとも500MbのRAM空き容量（1Gb以上を推奨）。
ハードディスク容量	Dr.Web for UNIX Internet Gateways ディレクトリのあるボリュームに1GB以上の空きディスク容量。
オペレーティングシステム	**GNU/Linux**（カーネルバージョン2.6.37以降、**glibc**ライブラリ2.13以降を使用）、**FreeBSD**（バージョン10.3以降）。 ⚠ オペレーションシステムは**PAM**認証メカニズムをサポートする必要があります。 コンポーネントDr.Web Firewall for Linuxが正しく動作するためには、以下のオプションを指定してOSカーネルが構築されている必要があります。 • *CONFIG_NETLINK_DIAG, CONFIG_INET_TCP_DIAG;* • *CONFIG_NF_CONNTRACK_IPV4, CONFIG_NF_CONNTRACK_IPV6, CONFIG_NF_CONNTRACK_EVENTS* • *CONFIG_NETFILTER_NETLINK_QUEUE, CONFIG_NETFILTER_NETLINK_QUEUE_CT, CONFIG_NETFILTER_XT_MARK.* 必要なオプションの組み合わせは、使用するOSのディストリビューションキットによって異なります。

コンポーネント	要件
	テスト済みのオペレーティングシステムのディストリビューションは以下のとおりです。
その他	以下の有効なネットワーク接続が必要です。 • ウイルスデータベースとDr.Web for UNIX Internet Gatewaysコンポーネントの更新を可能にする有効なインターネット接続。 • 集中管理モードで動作している場合は、ローカルネットワーク上のサーバーへの接続で十分です。インターネットへの接続は必要ありません。

Dr.Web for UNIX Internet Gatewaysを正しく動作させるために、以下のポートを開いてください。

機能	方向	ポート番号
更新を受け取るため	送信	80
Dr.Web Cloudサービスに接続するため	送信	2075（UDPのポート番号を含む）

テスト済みオペレーティングシステムのディストリビューション

Dr.Web for UNIX Internet Gatewaysは以下のディストリビューションでの動作が確認されています。

- **GNU/Linux**:

GNU/Linuxディストリビューション名	バージョン	プラットフォーム
Astra Linux Special Edition (Smolensk)	1.5、1.6	x86_64
ALT Linuxサーバー	9	ARM64
ALT Linuxワークステーション	9	ARM64
Debian	7.11、8.10、9.3	x86_64
Fedora	27～29	x86、x86_64
Red Hat Enterprise Linux	7.4	x86_64
CentOS	6.9、7.7、8	x86、x86_64、ARM64
SUSE Linux Enterprise Server	11 SP4、12 SP3	x86_64
Ubuntu	14.04、16.04、18.04	x86_64、ARM64

上記の要件を満たすその他の**GNU/Linux**ディストリビューションについては、Dr.Web for UNIX Internet Gatewaysとの互換性テストは行われていませんが、対応している場合があります。互換性の問題が発生した場合は、テクニカルサポートにお問い合わせください。

ARM64アーキテクチャの場合、次の**GNU/Linux**ディストリビューションの互換性がテストされています。Ubuntu 18.04、CentOS 7.7、ALT Linux Workstation 9、ALT Linux Server 9。他のディストリビューションはまだテストされていませんが、サポートされる可能性があります。互換性の問題が発生した場合は、テクニカルサポートにお問い合わせください。

- **FreeBSD**:

バージョン	プラットフォーム
11.2、12	x86、x86_64

FreeBSD OSの場合、Dr.Web for UNIX Internet Gatewaysはユニバーサルパッケージからのみインストールできます。

追加のパッケージとコンポーネント

Dr.Web for UNIX Internet Gatewaysでは、追加のパッケージやOSコンポーネントをインストールする必要はありません（保護されたサーバーソフトウェアを除く。以下を参照）。

コマンドラインでDr.Web for UNIX Internet Gatewaysを便利に操作するために、使用しているコマンドシェルでコマンド自動補完機能を有効にできます（無効になっている場合）。

追加のパッケージやコンポーネントのインストールに問題が発生した場合は、お使いのOSディストリビューションのマニュアルを参照してください。

免責事項

- SpIDer Gateは、OSにインストールされている他のファイアウォール（**SUSE Linux Enterprise Server** OSの**Shorewall**および**SuseFirewall2**、**Fedora** OS、**CentOS**、**RedHat Enterprise Linux**の**FirewallD**など）と*競合する可能性があります*。競合の兆候は、コードx109のSpIDer Gateのエラーに関するメッセージ、またはコードx102のDr.Web Firewall for Linuxのエラーに関するメッセージです。競合を解決する方法は、それぞれ「既知のエラー」セクションのエラーx109とx102で説明されています。
- 使用しているOSに1.4.15より前のバージョンの**NetFilter**が含まれている場合、SpIDer Gateは正しく動作しない可能性があります。この問題は、**NetFilter**の内部エラーに関連しており、SpIDer Gateを無効にすると、ネットワーク接続が切断され、再確立できなくなります。この問題が発生した場合は、バージョン1.4.15以降の**NetFilter** を含むOSにアップグレードすることをお勧めします。問題を解決する方法は、セクション「既知のエラーの説明」で説明されています。

サポート対象のHTTPプロキシサーバー

HTTPプロキシサーバーとの統合には、インストールおよび設定済みのHTTPプロキシサーバー**Squid** 3.0以降が必要です。**Squid**はICAPのサポートを有効化（`--enable-icap-client`オプションを指定してコンパイル）して構築される必要があります。

インターネットバリアと透過プロキシのモードでは、WebサーバーとHTTPプロキシサーバーに対する要件はありません。

インターネットバリアと透過プロキシモードは**GNU/Linux**でのみ動作します。

セキュリティサブシステムとの互換性

デフォルトでは、Dr.Web for UNIX Internet Gatewaysは **SELinux** をサポートしていません。また、Dr.Web for UNIX Internet Gatewaysは、強制アクセスモデルを使用する **GNU/Linux** システム（たとえば、ユーザーとファイルに異なる特権レベルを付与する **PARSEC** 強制アクセスサブシステムの備わったシステムなど）では機能が制限されたモードで動作します。

SELinuxを持つシステム（およびその他の強制アクセスモデルを使用するシステム）にDr.Web for UNIX Internet Gatewaysのインストールが必要な場合、Dr.Web for UNIX Internet Gatewaysの全機能が動作するように、セキュリティサブシステムの追加設定を実行する必要があります。詳細は、セキュリティサブシステムを設定するのセクションを参照してください。

ライセンス

Dr.Web for UNIX Internet Gatewaysを使用する権限は、Doctor Web社またはそのパートナーから購入した*ライセンス*によって付与されます。ユーザー権限を特定するライセンスパラメータは、Dr.Web for UNIX Internet Gatewaysのインストール中にユーザーが同意するライセンス契約（https://license.drweb.com/agreement/を参照）に従って設定されます。ライセンスには、ユーザーとベンダーに関する情報の他、以下を含む購入した製品の使用パラメータも含まれています。

- ユーザーに対してライセンスされたコンポーネントのリスト
- Dr.Web for UNIX Internet Gatewaysライセンスの有効期間
- その他の制限（購入した製品を使用できるコンピューターの台数など）。

評価目的のために、ユーザーは*試用期間*を有効化することもできます。有効化が成功すると、ユーザーは、有効化した試用期間全体にわたってDr.Web for UNIX Internet Gatewaysの全機能を使用できます。

各Doctor Web製品ライセンスには、ユーザーのコンピューターに保存されている特別なファイルに関連付けられた一意のシリアル番号があります。このファイルは、ライセンスパラメータに従ってコンポーネントの動作を制限し、*ライセンス*キーファイルと呼ばれます。試用期間が有効になると、*デモ*キーファイルと呼ばれる特別なキーファイルが自動的に生成されます。

コンピューターでライセンスまたは試用期間が有効になっていない場合、Dr.Web for UNIX Internet Gatewaysコンポーネントはブロックされます。さらに、ウイルスデータベースとコンポーネントの更新プログラムをDoctor Web更新サーバーからダウンロードすることもできません。ただし、企業またはインターネットサービスプロバイダーによって管理されるアンチウイルスネットワークの一部として製品を集中管理サーバーに接続することで、Dr.Web for UNIX Internet Gatewaysを有効化することができます。この場合、アンチウイルスの動作と更新は集中管理サーバーによって管理されます。

インストールとアンインストール

このセクションでは、Dr.Web for UNIX Internet Gatewaysバージョン11.1をインストールおよびアンインストールする方法について説明します。また、最新の更新を入手する方法や、Dr.Web for UNIX Internet Gatewaysの以前のバージョンがすでにコンピューターにインストールされている場合に新しいバージョンにアップグレードする手順も記載されています。

さらに、Dr.Web for UNIX Internet Gatewaysコンポーネントのカスタムインストールとアンインストール手順（Dr.Web for UNIX Internet Gatewaysの動作中に生じたエラーの解決方法や、機能セットを限定してインストールする方法など）、Dr.Web for UNIX Internet Gatewaysのインストールと動作に必要になる可能性がある高度なセキュリティサブシステムの設定（**SELinux**など）についてもご確認いただけます。

- Dr.Web for UNIX Internet Gatewaysをインストールする。
- Dr.Web for UNIX Internet Gatewaysをアップグレードする。
- Dr.Web for UNIX Internet Gatewaysをアンインストールする。
- セキュリティサブシステムを設定する。
- 追加情報：
  - Dr.Web for UNIX Internet Gatewaysパッケージとファイル。
  - コンポーネントのカスタムインストールとアンインストール。

これらの手順を実行するには、スーパーユーザー権限（*root* ユーザーの権限）が必要です。権限を昇格するには、**su** コマンド（カレントユーザーを変更する）または **sudo** コマンド（指定されたコマンドを別のユーザーの権限で実行する）を使用します。

Dr.Web for UNIX Internet Gatewaysと他社のアンチウイルス製品との互換性は*保証されません*。1台のシステム上に2つのアンチウイルスがインストールされることで、*OSのエラーを引き起こし*、*重要なデータが失われる*可能性があります。Dr.Web for UNIX Internet Gatewaysをインストールする前に、他社アンチウイルス製品をコンピューターから削除することが*強く推奨されます*。

お使いのコンピューターに他のDr.Webアンチウイルス製品がユニバーサルパッケージ（`.run`）から*すでにインストールされていて*、さらに別のDr.Webアンチウイルス製品をインストールする場合（たとえば、ユニバーサルパッケージからDr.Web for Linuxをインストールしていて、さらにDr.Web for UNIX Internet Gatewaysをインストールする場合など）、インストールされている製品のバージョンがインストールするDr.Web for UNIX Internet Gatewaysのバージョンと*同じ*であることを確認してください。新しくインストールする製品のバージョンがインストールされている製品のものよりも新しい場合、インストール*前*に、インストールされているDr.Web for UNIX Internet Gatewaysを新しくインストールする製品のバージョンにアップグレードする必要があります。

FreeBSD OSの場合、Dr.Web for UNIX Internet Gatewaysはユニバーサルパッケージからのみインストールできます。

Dr.Web for UNIX Internet Gatewaysをインストールする

Dr.Web for UNIX Internet Gatewaysをインストールするには、以下の手順のいずれか1つを行います。

1. Doctor Webの公式Webサイトから、UNIXシステム用のユニバーサルパッケージを含むインストールファイルをダウンロードします。パッケージには、1つのインストーラが含まれています（インストールプログラムはコマンドラインモード用に開発されているため、グラフィカルデスクトップモードで使用するには、使用可能なターミナルエミュレーターが必要です）。
2. Doctor Webの対応するパッケージリポジトリからネイティブパッケージをダウンロードします。

> **FreeBSD** OSの場合、Dr.Web for UNIX Internet Gatewaysはユニバーサルパッケージからのみインストールできます。

インストール中（ユニバーサル.runパッケージまたはパッケージマネージャーを使用してネイティブパッケージから）、インストール結果を含むメッセージが`root@localhost`に送信されます。

上記のいずれかの方法でDr.Web for UNIX Internet Gatewaysをインストールした後、そのコンポーネントに利用可能な修正があるか、新しいDr.Web for UNIX Internet Gatewaysバージョンがリリースされている場合は、アンインストールまたは更新できます。必要に応じて、Dr.Web for UNIX Internet Gatewaysを正しく動作させるために、**GNU/Linux**のセキュリティサブシステムを設定することもできます。個々のコンポーネントの機能に問題がある場合は、Dr.Web for UNIX Internet Gatewaysをアンインストールせずに、該当するコンポーネントのカスタムインストールおよびアンインストールを実行できます。

選択したDr.Web for UNIX Internet Gatewaysのインストール方法に関係なく、インストールが完了したら、ライセンスを有効化して、受け取ったキーファイルをインストールする必要があります。さらに、Dr.Web for UNIX Internet Gatewaysを集中保管理サーバーに接続することもできます。詳細は、ライセンスを参照してください。これを行わない場合、アンチウイルス保護機能が無効になります。さらに、場合によっては、開始するセクションで説明されているように、インストールしたDr.Web for UNIX Internet Gatewaysの基本機能をカスタマイズする必要があります。

ユニバーサルパッケージをインストールする

Dr.Web for UNIX Internet Gatewaysユニバーサルパッケージは、`drweb-`*<version>*`-av-igw-`*<OS>*`-`*<platform>*`.run`という名前のインストールファイルとして提供されます。ここで、*<OS>*は**UNIX**ベースのオペレーティングシステムのタイプ、*<Platform>*はDr.Web for UNIX Internet Gatewaysが対象としているプラットフォーム（32ビットプラットフォームは`x86`、64ビットプラットフォームは`amd64`）です。例:

```
drweb-11.1-av-igw-linux-x86.run
```

上のフォーマットに該当するインストールファイルの名前は、このセクション以降では*<file_name>*`.run`と表記します。

Dr.Web for UNIX Internet Gatewaysのコンポーネントをインストールするには以下の手順を行ってください。

1. ユニバーサルパッケージを含むインストールファイルがない場合は、Doctor Webの公式Webサイト（https://download.drweb.com/）からダウンロードしてください。
2. インストールファイルをコンピューターのハードディスクドライブに保存します。
3. 次のコマンド（例）を使用してアーカイブを実行できるようにします。

```
# chmod +x <file_name>.run
```

4. 以下のコマンドを使用してアーカイブを実行します。

```
# ./<file_name>.run
```

ファイルプロパティ（パーミッション）の変更とファイルの実行は、グラフィカルシェルの標準的なファイルマネージャーを使用することも可能です。

これにより、アーカイブの整合性チェックが実行され、その後にアーカイブファイルが一時ディレクトリに展開され、インストールプログラムが開始されます。ユーザーがroot権限を持っていない場合、インストールプログラムはそのユーザーにrootパスワードを要求して権限を昇格させようとします（**sudo**が使用されます）。この試みが失敗した場合、インストールプロセスは停止します。

ファイルシステム内の一時ディレクトリへのパスに、展開されるファイル用の十分な空き容量がない場合、インストールプロセスが停止し、対応するメッセージが表示されます。この場合は、TMPDIRシステム環境変数の値を、十分な空き容量があるディレクトリを指すように変更し、インストールをやり直してください。`-- target`オプションを使用することもできます。

その後、コマンドラインモード用のインストーラが自動的に起動されます（グラフィカルデスクトップ環境で実行するには、ターミナルエミュレーターが必要です）。

5. インストーラの指示に従ってください。
6. 次のコマンドを実行することで、インストールプログラムをサイレントモードで実行することもできます。

```
# ./<file_name>.run -- --non-interactive
```

この場合、インストールプログラムはサイレントモードで起動され、ユーザーインターフェースなしで動作します（コマンドラインモードで表示されるダイアログも表示されません）。

注意事項

- このオプションを使用すると、Dr.Webライセンス契約の規定に*同意する*ことになります。ライセンス契約のテキストは、`/opt/drweb.com/share/doc/LICENSE`ファイルにあります。ファイル拡張子は、ライセンス契約の言語を示しています。*ライセンス*ファイルに拡張子がない場合、Dr.Webライセンス契約は英語で記述されています。ライセンス契約の条件に*同意しない*場合は、インストール後にDr.Web for UNIX Internet Gatewaysをアンインストールしてください。
- アンインストールプログラムをサイレントモードで実行するには管理者（root）権限が必要です。権限を昇格するには、**su** および **sudo** コマンドを使用できます。

使用している**GNU/Linux**ディストリビューションが**SELinux**を備えている場合、インストールプロセスがセキュリティサブシステムによって中断される可能性があります。このような場合は、**SELinux**を*Permissive*モードに設定してください。これを行うには、次のコマンドを入力します。

```
# setenforce 0
```

次に、インストーラを再起動させます。インストールが完了した後、製品コンポーネントの正常な動作を可能にするよう **SELinux** セキュリティポリシー を設定します。

<opt_dir>、*<etc_dir>*、*<var_dir>*の表記規則の詳細は、はじめにを参照してください。

インストールプロセスが完了すると、展開されたインストールファイルがすべて削除されます。

<file_name>.runファイル（インストール元のファイル）を保存しておくことを推奨します。これにより、バージョンの更新を必要とせずにDr.Web for UNIX Internet Gatewaysやそのコンポーネントを再インストールすることが可能になります。

コマンドラインからインストールする

コマンドラインベースのインストールプログラムを起動すると、製品をインストールするように促すメッセージが表示されます。

1. インストールを開始するには、「Do you want to continue?」に対して「*Yes*」または「*Y*」を入力します。インストーラを終了するには、「*No*」または「*N*」と入力します。この場合、インストールはキャンセルされます。
2. その後、画面に表示されているDr.Webライセンス契約の規約を確認する必要があります。Enterキーを押してテキストを1行ずつ下にスクロールするか、またはSpaceキーを押してテキストを1画面ずつ下にスクロールします。ライセンス契約を上にスクロールするオプションはありません。
3. ライセンス契約のテキストを読んだ後、規約に同意するように求められます。ライセンス契約に同意する場合は、*Yes*または*Y*を入力します。同意しない場合は、*No*または*N*を入力します。後者の場合、インストーラは終了します。
4. ライセンス契約の規約に同意すると、インストールが自動的に開始されます。手順の実行中、インストールされるDr.Webコンポーネントの一覧を含むインストールプロセスに関する情報が画面に表示されます。
5. インストールが正常に完了すると、Dr.Web for UNIX Internet Gatewaysの動作を管理する方法を通知するメッセージが表示されます。

エラーが発生した場合は、エラーについて説明するメッセージが画面に表示された後、インストーラが終了します。インストールがエラーによって失敗した場合、エラーの原因を取り除き、インストールを再度開始してください。

リポジトリからインストールする

Dr.Web for UNIX Internet Gatewaysのネイティブパッケージはhttps://repo.drweb.com/にあるDr.Webの公式リポジトリに保存されています。お使いのOSのパッケージマネージャーが使用するリポジトリのリストにDr.Webリポジトリを追加すると、OSのリポジトリから他のプログラムをインストールするのと同じようにネイティブパッケージから製品をインストールできます。必要な依存関係は自動的に解決されます。

以下で説明するコマンド（リポジトリの追加、デジタル署名キーのインポート、パッケージのインストールと削除に使用するコマンド）はすべて、スーパーユーザー（**root**）管理者権限で実行する必要があります。管理者権限を昇格するには、（現在のユーザーを変更する）**su**コマンド、または（他のユーザーの管理者権限で指定したコマンドを実行する）**sudo**コマンドを使用します。

FreeBSD OSの場合、Dr.Web for UNIX Internet Gatewaysはユニバーサルパッケージからのみインストールできます。

以下のOS（パッケージマネージャー）の手順を参照してください。

- Debian, Mint, Ubuntu (apt)

- ALT Linux, PCLinuxOS (apt-rpm)
- Mageia, OpenMandriva Lx (urpmi)
- Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf)
- SUSE Linux (zypper)

Debian、Mint、Ubuntu（apt）

1. これらOS用のリポジトリは Doctor Web によって電子署名されています。リポジトリにアクセスするには、以下のコマンドを実行することで、デジタル署名キーをインポートし、パッケージマネージャーストレージに追加します。

```
# apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys
8C42FC58D8752769
```

2.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追加します。

```
deb http://repo.drweb.com/drweb/debian 11.1 non-free
```

1および2の項目は、リポジトリから特別なDEBパッケージ（https://repo.drweb.com/drweb/drweb-repo11.1.deb）をダウンロードしてインストールすることでも実行できます。

3. リポジトリから 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用します。

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

代わりのパッケージマネージャー（**Synaptic** または **aptitude** など）を使用して製品をインストールすることもできます。パッケージの競合が発生した場合は、それを解決するために **aptitude** などの代わりのマネージャーを使用することが推奨されます。

ALT Linux、PCLinuxOS（apt-rpm）

1.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追加します。

```
rpm http://repo.drweb.com/drweb/altlinux 11.1/<arch> drweb
```

*<arch>*は、使用されるパッケージ構造を表します。

- **32-bit**バージョン：`i386`
- **64-bit**バージョン：`x86_64`
- **ARM64**アーキテクチャの場合：`aarch64`

2. リポジトリから 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用します。

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

代わりのパッケージマネージャー（**Synaptic** または **aptitude** など）を使用して製品をインストールすることもできます。

Mageia、OpenMandriva Lx（urpmi）

1.以下のコマンドを使用してリポジトリを接続します。

```
# urpmi.addmedia drweb https://repo.drweb.com/drweb/linux/11.1/<arch>/
```

*<arch>*は、使用されるパッケージ構造を表します。

- **32-bit**バージョン：`i386`
- **64-bit**バージョン：`x86_64`

2. リポジトリから 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用します。

```
# urpmi drweb-internet-gateways
```

代わりのパッケージマネージャー（**rpmdrake** など）を使用して製品をインストールすることもできます。

Red Hat Enterprise Linux、Fedora、CentOS（yum、dnf）

1. 以下のコンテンツが含まれた `drweb.repo` ファイルを `/etc/yum.repos.d` ディレクトリに追加します。

```
[drweb]
name=DrWeb-11.1
baseurl=https://repo.drweb.com/drweb/linux/11.1/$basearch/
gpgcheck=1
enabled=1
gpgkey=https://repo.drweb.com/drweb/drweb.key
```

echo などのコマンドを使用して上のコンテンツをファイルにロギングし、出力をリダイレクトする場合は、`$` 記号をエスケープする必要があります（`\$`）。

項目 1は、リポジトリから特別なRPMパッケージ（https://repo.drweb.com/drweb/drweb-repo11.1.rpm）をダウンロードしてインストールすることでも実行できます。

2. リポジトリから 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用します。

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

Fedora のバージョン22以降では、マネージャー **yum** の代わりに **dnf** を使用することが推奨されます。例:

```
# dnf install drweb-internet-gateways
```

代わりのパッケージマネージャー(**PackageKit** または **Yumex** など)を使用して製品をインストールすることもできます。

SUSE Linux(zypper)

1. リポジトを追加するには、以下のコマンドを使用します。

```
# zypper ar https://repo.drweb.com/drweb/linux/11.1/\$basearch/ drweb
```

2. リポジトリから 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用します。

```
# zypper refresh
# zypper install drweb-internet-gateways
```

代わりのパッケージマネージャー(**YaST** など)を使用して製品をインストールすることもできます。

Dr.Web for UNIX Internet Gatewaysをアップグレードする

Dr.Web for UNIX Internet Gatewaysには2つの更新モードがあります。

1. 現在の製品バージョンに対してリリースされたパッケージやコンポーネントの更新を入手する。通常、このような更新ではエラー修正やコンポーネント機能の軽微な改良が行われています。
2. Dr.Web for UNIX Internet Gatewaysの新しいバージョンにアップグレードする。このアップグレードオプションは、お使いのDr.Web for UNIX Internet Gatewaysの新しいバージョンをDoctor Webがリリースし、それに新しい機能が備わっている場合に使用されます。

最新のアップデートを入手する

該当するセクションに記載されている方法を使用してDr.Web for UNIX Internet Gatewaysをインストールすると、パッケージマネージャーは自動的にDr.Webパッケージリポジトリに接続します。

- インストールがユニバーサルパッケージ(ファイル.run)から実行され、システムでDEBパッケージが使用されていて(たとえば、**Debian**、**Mint**、**Ubuntu**などのOS)、OS(**FreeBSD**)にパッケージマネージャーがない場合、Dr.Webパッケージの動作には、それぞれのバージョンのパッケージマネージャー**zypper**が使用されます。これはDr.Web for UNIX Internet Gatewaysのインストール時に自動的にインストールされます。

  このマネージャーが含まれる更新されたDr.Webパッケージを入手してインストールするには、*<opt_dir>*/binディレクトリ(**GNU/Linux**—/opt/drweb.com/bin)へ行き、次のコマンドを実行します。

```
# ./zypper refresh
# ./zypper update
```

FreeBSD OS v.11.x for amd64では、更新に**zypper**マネージャーを使用すると、リポジトリアップデートエラーが発生する場合があります。この場合、compat10x-amd64サポートパッケージをインストールして、再試行してください。

パッケージをインストールするには、次のコマンドを使用します。

```
# pkg install compat10x-amd64
```

- それ以外の場合は、お使いのOSで使用されているパッケージマネージャーを更新するコマンドを使用します。例:
  - **Red Hat Enterprise Linux** と**CentOS**では、**yum**コマンドを使用します。
  - **Fedora**では、**yum**または**dnf**コマンドを使用します。
  - **SUSE Linux**では、**zypper**コマンドを使用します。
  - **Mageia**と**OpenMandriva Lx**では、**urpmi**コマンドを使用します。
  - **Alt Linux**、**PCLinuxOS**、**Debian**、**Mint**、**Ubuntu** では、**apt-get** コマンドを使用します。

また、お使いのOS用に開発された別のパッケージマネージャーを使用することもできます。必要に応じて、使用しているパッケージマネージャーのマニュアルを参照してください。

新しいDr.Web for UNIX Internet Gatewaysバージョンがリリースされると、そのコンポーネントを含むパッケージは、新しいバージョンに対応するDr.Webリポジトリのセクションに配置されます。この場合、更新にはパッケージマネージャーを新しいDr.Webリポジトリセクションに切り替える必要があります(新しいバージョンにアップグレードするを参照)。

新しいバージョンにアップグレードする

このセクションの内容:

注意事項

新しいバージョンにアップグレードする前に、サーバーが新しいバージョンのシステム要件を満たしていることを確認してください。これには、必要なプログラムがインストールされていることを含みます。

お使いのバージョンのDr.Web for UNIX Internet Gatewaysは、製品のインストール時と同じ方法でアップグレードする必要があります。

- Dr.Web for UNIX Internet Gatewaysの現在のバージョンがリポジトリからインストールされている場合、アップグレードではリポジトリからのプログラムパッケージを更新する必要があります。
- Dr.Web for UNIX Internet Gatewaysの現在のバージョンがユニバーサルパッケージからインストールされている場合、Dr.Web for UNIX Internet Gatewaysをアップグレードするには、新しいバージョンの製品を含んでいる別のユニバーサルパッケージをインストールする必要があります。

製品バージョンのインストール方法を特定するには、Dr.Web for UNIX Internet Gatewaysの実行可能ディレクトリに`uninst.sh`のアンインストールスクリプトが含まれているかどうかを確認します。その場合、現在のバージョンはユニバーサルパッケージからインストールされています。それ以外の場合は、リポジトリからインストールされています。

FreeBSD OSの場合、Dr.Web for UNIX Internet Gatewaysはユニバーサルパッケージからのみインストールできます。

インストールに使用した方法でDr.Web for UNIX Internet Gatewaysを更新できない場合は、現在のバージョンのDr.Web for UNIX Internet Gatewaysをアンインストールしてから、何らかの方法で新しいバージョンをインストールしてください。以前のバージョンのDr.Web for UNIX Internet Gatewaysのインストールおよびアンインストール手順は、バージョン11.1向けの本マニュアルにあるインストールおよびアンインストールと同じです。追加情報については、最新バージョンのDr.Web for UNIX Internet Gatewaysのユーザーマニュアルを参照してください。

現在のバージョンのDr.Web for UNIX Internet Gatewaysが集中管理モードで動作している場合は、使用されている集中管理サーバーのアドレスを記録することをお勧めします。たとえば、バージョン6.0.2以降のDr.Web for UNIX Internet Gatewaysのアドレスを確認するには、次のコマンドを使用できます。

```
$ drweb-ctl appinfo
```

このコマンドの結果、次のような行が出力されます。

```
ESAgent; <PID>; RUNNING 1; Connected <address>, on-line
```

*<address>*の部分を保存します（`tcp://`*<IP address>*`:`*<port>*のようになっています。例：`tcp://10.20.30.40:1234`）。また、サーバー証明書ファイルを保存することをお勧めします。

現在使用している接続パラメータを調べる際に問題が発生した場合は、お使いのバージョンのDr.Web for UNIX Internet Gateways用管理者マニュアルをご確認いただくか、アンチウイルスネットワーク管理者までお問い合わせください。

アップグレードのためにユニバーサルパッケージをインストールする

ユニバーサルパッケージからDr.Web for UNIX Internet Gateways 11.1をインストールします。自動更新が不可能な場合は、新しいバージョンのインストール中に、コンピューターにインストールされているDr.Web for UNIX Internet Gatewaysの古いバージョンのコンポーネントを自動的に削除するメッセージが表示されます。

更新中に、インストールされたDr.Web for UNIX Internet Gatewaysバージョンを削除する必要があり、Dr.Webの複数のサーバー製品（ファイルサーバー用、メールサーバー用、インターネットゲートウェイ用などの製品）がお使いのサーバーに*一緒に*インストールされている場合、アップグレードされない他のサーバー製品を完全に機能させるために（すなわち、ファイルサーバー用およびメールサーバー用の製品、を維持するために）、削除対象として以下のパッケージ*のみ*を選択する必要があります。

- `drweb-internet-gateways-doc`
- `drweb-icapd-web`
- `drweb-icapd`

リポジトリからアップグレードする

Dr.Webバージョン6.0.2の複数のサーバー製品がお使いのサーバーに*一緒に*インストールされている場合（ファイルサーバー用製品、メールサーバー用製品、インターネットゲートウェイ用製品がインストールされている場合など）、Dr.Web for UNIX Internet Gateways 6.0.2をリポジトリからバージョン11.1にアップグレードすることは*できません*。この場合は、新しいバージョンのDr.Web for UNIX Internet Gatewaysを別のマシンにインストールしてください。

Doctor Webリポジトリからインストールされた現在のバージョンのDr.Web for UNIX Internet Gatewaysをアップグレードするには、必要なパッケージのタイプに応じて、次のいずれかの操作を行います。

- **RPMパッケージ（yum、dnf）。**
  1. 使用するリポジトリを変更します（現在のバージョンのパッケージリポジトリから11.1パッケージリポジトリへ）。

     リポジトリからインストールするセクションで、11.1パッケージが保存されているリポジトリの名前を確認できます。詳しいリポジトリの変更方法については、お使いのOSディストリビューションのヘルプガイドを参照してください。

  2. 以下のコマンドを使用して新しいバージョンをインストールします。

     ```
     # yum update
     ```

     または、マネージャー **dnf** を使用している場合（バージョン22以前の **Fedora** など）は、以下のコマンドを使用します。

     ```
     # dnf update
     ```

     パッケージの更新中にエラーが発生した場合は、Dr.Web for UNIX Internet Gatewaysをアンインストールして、再度インストールしてください。必要に応じて、セクションリポジトリからインストールしたDr.Web for UNIX Internet Gatewaysをアンインストールするおよびリポジトリからインストールする（使用しているOSおよびパッケージマネージャーの項目）を参照してください。

- **DEBパッケージ（apt-get）。**
  1. 使用するリポジトリを変更します（現在のバージョンのパッケージリポジトリから11.1パッケージリポジトリへ）。
  2. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysパッケージをアップグレードします。

```
# apt-get update
# apt-get dist-upgrade
```

Ubuntu 14.04（64ビット版）OSの場合、**apt-get** `dist-upgrade` コマンドが失敗する場合がありますので注意してください。この場合は、**aptitude** パッケージマネージャーを使用してください（製品をアップグレードするには、**aptitude** `dist-upgrade` コマンドを実行します）。

キーファイルの転送

選択したDr.Web for UNIX Internet Gatewaysのアップグレード方法に関係なく、すでにあるライセンスキーファイル（お持ちの場合）は自動的に転送され、新しいバージョンで必要な場所にインストールされます。

キーファイルの自動インストール中に問題が発生した場合は、手動でインストールできます。

有効なライセンスキーファイルを紛失した場合は、テクニカルサポートに連絡してください。

集中管理サーバーとの接続を復元する

可能であれば、アップグレード後に集中管理サーバーへの接続が自動的に復元されます（アップグレード前に製品が集中管理サーバーに接続されていた場合）。接続が自動的に復元されなかった場合は、アップグレードしたDr.Web for UNIX Internet Gatewaysのアンチウイルスネットワークへの接続を再度確立するために、次のコマンドを実行します。

```
$ drweb-ctl esconnect <アドレス> --Certificate <証明書ファイルへのパス>
```

接続処理に問題が発生した場合は、アンチウイルスネットワークの管理者までお問い合わせください。

Dr.Web for UNIX Internet Gatewaysをアンインストールする

Dr.Web for UNIX Internet Gateways をインストールした方法に応じて、次のいずれかの方法で製品をアンインストールできます。

1. アンインストーラを起動して、ユニバーサルパッケージをアンインストールします。
2. システムのパッケージマネージャーを使用して、Doctor Webのリポジトリからインストールしたパッケージをアンインストールします。

ユニバーサルパッケージをアンインストールする

UNIXシステム向けのユニバーサルパッケージからインストールされたDr.Web for UNIX Internet Gatewaysは、コマンドラインからアンインストールできます（このオプションでは、グラフィカルデスクトップ環境を使用している場合、端末エミュレーターが必要です）。

アンインストールツールはDr.Web for UNIX Internet Gatewaysだけでなく、コンピューターにインストールされている *他のすべての* Dr.Web製品をアンインストールすることに注意してください。

Dr.Web for UNIX Internet Gateways以外の他のDr.Web製品がコンピューターにインストールされている状況でDr.Web for UNIX Internet Gatewaysのみを削除するには、自動削除ツールを実行する代わりにコンポーネントのカスタムインストールとアンインストールの手順を使用します。

コマンドラインからDr.Web for UNIX Internet Gatewaysをアンインストールする

アンインストールツールは、*<opt_dir>*/binディレクトリ(**GNU/Linux**では/opt/drweb.com/bin)にあるuninst.shスクリプトによって起動されます。Dr.Web for UNIX Internet Gatewaysのアンインストール手順については、セクションコマンドラインからアンインストールする

次のコマンドを実行することで、アンインストールツールをサイレントモードで起動することもできます。

```
# env DRWEB_NON_INTERACTIVE=yes /opt/drweb.com/bin/uninst.sh
```

この場合、アンインストールツールはサイレントモードで実行され、ユーザーインターフェースなしで動作します(コマンドラインモードのプログラムダイアログを含む)。アンインストールツールをサイレントモードで実行するにはroot権限が必要です。権限を昇格するには、**su** および **sudo** コマンドを使用できます。

コマンドラインからアンインストールする

コマンドラインベースの削除プログラムが起動すると、製品をアンインストールするメッセージがコマンドラインに表示されます。

1. 削除を開始するには、「Do you want to continue?」に対して「*Yes*」または「*Y*」を入力します。アンインストーラを終了するには、「*No*」または「*N*」と入力します。この場合、Dr.Web製品の削除はキャンセルされます。
2. アンインストールを確定すると、インストールされているすべてのDr.Web製品の自動アンインストールが開始されます。この手順の間、アンインストールのプロセスに関する情報が画面に表示され、アンインストールログに記録されます。
3. プロセスが完了すると、アンインストールプログラムは自動的に終了します。

リポジトリからインストールしたDr.Web for UNIX Internet Gatewaysをアンインストールする

以下のOS(パッケージマネージャー)の手順を参照してください。

- Debian, Mint, Ubuntu (apt)
- ALT Linux, PCLinuxOS (apt-rpm)
- Mageia, OpenMandriva Lx (urpmi)
- Red Hat Enterprise Linux, Fedora, CentOS (yum, dnf)
- SUSE Linux (zypper)

パッケージのアンインストールにおいて、以下に記載されているすべてのコマンドは、スーパーユーザー（**root**）管理者権限を必要とします。管理者権限を昇格するには、（現在のユーザーを変更する）**su**コマンド、または（他のユーザーの管理者権限で指定したコマンドを実行する）**sudo**コマンドを使用します。

Debian、Mint、Ubuntu（apt）

Dr.Web for UNIX Internet Gateways のルートメタパッケージをアンインストールするには、以下のコマンドを入力します。

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

インストールされているすべてのDr.Webパッケージをアンインストールするには、以下のコマンドを入力します（一部のOSでは、「*」記号をエスケープする必要があります：「*」）。

```
# apt-get remove drweb*
```

不要になったすべてのパッケージを自動的にアンインストールするには、以下のコマンドを入力します。

```
# apt-get autoremove
```

apt-get コマンドを使用したアンインストールには以下の特徴がありますので注意してください。

1. 最初のコマンドのケースでは、`drweb-internet-gateways`パッケージのみをアンインストールします。このパッケージの依存関係を解決するのに自動的にインストールされた可能性のある他のパッケージはシステムに残ります。
2. 2番目のコマンドのケースでは、名前が「`drweb`」（Dr.Web製品名の標準的接頭辞）で始まるすべてのパッケージをアンインストールします。このコマンドはDr.Web for UNIX Internet Gatewaysのパッケージだけでなく、この接頭辞を持つパッケージをすべてアンインストールします。
3. 3番目のコマンドのケースでは、他のパッケージの依存関係を解決するために自動的にインストールされ、依存パッケージのアンインストールなどにより不要になったパッケージをすべてアンインストールします。このコマンドはDr.Web for UNIX Internet Gatewaysのパッケージだけでなく、使用されていないすべてのパッケージをアンインストールします。

代わりのマネージャー（**Synaptic** または **aptitude** など）を使用してパッケージをアンインストールすることもできます。

ALT Linux、PCLinuxOS（apt-rpm）

この場合、Dr.Web for UNIX Internet Gatewaysのアンインストールは、**Debian**および**Ubuntu**上でのアンインストールと同じです（上を参照）。

代わりのマネージャー（**Synaptic** または **aptitude** など）を使用してパッケージをアンインストールすることもできます。

Mageia、OpenMandriva Lx（urpme）

Dr.Web for UNIX Internet Gateways をアンインストールするには、以下のコマンドを入力します。

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

不要になったすべてのパッケージを自動的にアンインストールするには、以下のコマンドを入力します。

```
# urpme --auto-orphans drweb-internet-gateways
```

urpme コマンドを使用したアンインストールには以下の特徴がありますので注意してください。

1. 最初のコマンドのケースでは、`drweb-internet-gateways`パッケージのみをアンインストールします。このパッケージの依存関係を解決するのに自動的にインストールされた可能性のある他のパッケージはシステムに残ります。
2. 2番目のコマンドのケースでは、`drweb-internet-gateways`パッケージと、他のパッケージの依存関係を解決するために自動的にインストールされ、依存パッケージのアンインストールなどにより不要になったすべてのパッケージをアンインストールします。このコマンドはDr.Web for UNIX Internet Gatewaysのパッケージだけでなく、使用されていないすべてのパッケージをアンインストールします。

代わりのマネージャー（**rpmdrake**など）を使用してパッケージをアンインストールすることもできます。

Red Hat Enterprise Linux、Fedora、CentOS（yum、dnf）

インストールされているすべてのDr.Webパッケージをアンインストールするには、以下のコマンドを入力します（一部のOSでは、「`*`」記号をエスケープする必要があります：「`\*`」）。

```
# yum remove drweb*
```

Fedora のバージョン22以降では、マネージャー **yum** の代わりに **dnf** を使用することが推奨されます。例：

```
# dnf remove drweb*
```

yum（**dnf**）コマンドを使用したアンインストールには以下の特徴がありますので注意してください。

このコマンドのケースでは、名前が「`drweb`」（Dr.Web製品名の標準的接頭辞）で始まるすべてのパッケージをアンインストールします。このコマンドはDr.Web for UNIX Internet Gatewaysのパッケージだけでなく、この接頭辞を持つパッケージをすべてアンインストールします。

代わりのマネージャー（**PackageKit** または **Yumex** など）を使用してパッケージをアンインストールすることもできます。

SUSE Linux（zypper）

Dr.Web for UNIX Internet Gateways をアンインストールするには、以下のコマンドを入力します。

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

インストールされているすべてのDr.Webパッケージをアンインストールするには、以下のコマンドを入力します（一部のOSでは、「*」記号をエスケープする必要があります：「*」）。

```
# zypper remove drweb*
```

zypper コマンドを使用したアンインストールには以下の特徴がありますので注意してください。

1. 最初のコマンドのケースでは、drweb-internet-gatewaysパッケージのみをアンインストールします。このパッケージの依存関係を解決するのに自動的にインストールされた可能性のある他のパッケージはシステムに残ります。
2. 2番目のコマンドのケースでは、名前が「drweb」（Dr.Web製品名の標準的接頭辞）で始まるすべてのパッケージをアンインストールします。このコマンドはDr.Web for UNIX Internet Gatewaysのパッケージだけでなく、この接頭辞を持つパッケージをすべてアンインストールします。

代わりのマネージャー（**YaST** など）を使用してパッケージをアンインストールすることもできます。

追加情報

Dr.Web for UNIX Internet Gatewaysパッケージとファイル

パッケージ

Dr.Web for UNIX Internet Gatewaysは以下のパッケージで構成されています。

パッケージ	コンテンツ
drweb-bases	ウイルスデータベースファイル
drweb-boost	ブーストライブラリ
drweb-clamd	Dr.Web ClamDコンポーネントのファイル
drweb-cloudd	Dr.Web CloudDコンポーネントのファイル
drweb-common	メインの設定ファイル - drweb.ini、メインライブラリ、ドキュメント、Dr.Web for UNIX Internet Gatewaysディレクトリの階層、製品設定とシステム環境に関する情報を収集するためのユーティリティ。 このパッケージのインストール中に、drwebという名前のユーザーとdrwebという名前のグループが作成されます。

パッケージ	コンテンツ
drweb-configd	Dr.Web ConfigDファイル
drweb-configure	Dr.Web for UNIX Internet Gateways設定用の補助ツールのファイル
drweb-ctl	Dr.Web Ctlファイル
drweb-documentation	HTMLフォーマットのDr.Web for UNIX製品ドキュメントファイル
drweb-dws	Webリソースカテゴリーのデータベースのファイル
drweb-engine	Dr.Web Virus-Finding Engineスキャンエンジンファイル
drweb-esagent	Dr.Web ES Agentコンポーネントのファイル
drweb-filecheck	Dr.Web File Checkerコンポーネントのファイル
drweb-internet-gateways-doc	PDFドキュメント
drweb-internet-gateways	Dr.Web for UNIX Internet Gatewaysのルートメタパッケージ
drweb-gated	SpIDer Gateコンポーネントのファイル
drweb-firewall	Dr.Web Firewall for Linuxコンポーネントのファイル
drweb-httpd	Dr.Web HTTPDコンポーネントと管理Webインターフェース（メタパッケージ）のファイル
drweb-httpd-bin	Dr.Web HTTPDコンポーネントのファイル
drweb-httpd-webconsole	管理Webインターフェースのファイル
drweb-icu	Unicodeサポートとインターナショナライゼーションのためのライブラリ
drweb-icapd	Dr.Web ICAPDコンポーネントのファイル
drweb-libs	メインライブラリファイル
drweb-lookupd	Dr.Web LookupDコンポーネントのファイル
drweb-lua	ネットワーク接続監視用に設計されたDr.Web for UNIX Internet Gatewaysコンポーネントによって使用されるLuaインタプリタのファイル
drweb-netcheck	Dr.Web Network Checkerコンポーネントのファイル
drweb-openssl	**OpenSSL**ライブラリ
drweb-protobuf	**Google Protobuf**ライブラリ
drweb-se	Dr.Web Scanning Engineコンポーネントのファイル
drweb-snmpd	Dr.Web SNMPDコンポーネントのファイル

パッケージ	コンテンツ
drweb-update	Dr.Web Updaterコンポーネントのファイル

セクションコンポーネントのカスタムインストールとアンインストールには、Dr.Web for UNIX Internet Gatewaysの典型的なタスクに対する解決策を提供するカスタムインストール用の典型的なコンポーネントセットがあります。

ファイル

Dr.Web for UNIX Internet Gatewaysのインストール後、その構成ファイルはファイルシステムの/opt、/etc、/var ディレクトリに置かれます。

使用されるディレクトリの構造

ディレクトリ	コンテンツ
• **GNU/Linux**の場合： /etc/init.d/ • **FreeBSD**の場合： /usr/local/etc/rc.d/	Dr.Web ConfigDデーモン用のdrweb-configdスクリプト。
<etc_dir>	drweb.ini設定ファイルとdrweb32.keyキーファイル。また、次が含まれます。
certs/	– 使用中の証明書のファイル。
<opt_dir>/	Dr.Web for UNIX Internet Gatewaysのメインディレクトリ。次が含まれます。
bin/	– 全製品のコンポーネントの実行ファイル（Dr.Web Virus-Finding Engineを除く）。
include/	– 使用中のライブラリのヘッダーファイル。
lib/	– 使用中のライブラリ。
man/	– システムヘルプファイル：**man**。
share/	– 次を含む補助製品ファイル。
doc/	▫ 製品ドキュメント（readmeファイル、ライセンス契約）、パッケージがすでにインストールされている場合は管理者ガイド）。
drweb-bases/	▫ Dr.Webのウイルスデータベースのファイル（インストール中に提供されたソースファイル）。
scripts/	▫ 補助スクリプトファイル。
<var_dir>/	以下を含む補助ファイルと一時ファイル：
bases/	– Dr.Webウイルスデータベースのファイル（更新バージョン）。

ディレクトリ	コンテンツ
cache/	– 更新のキャッシュ。
drl/	– 使用中の更新サーバーのリスト。
dws/	– Webリソースカテゴリーのデータベースのファイル。
lib/	– ダイナミックリンクライブラリとしてのDr.Web Virus-Finding Engineスキャンエンジン(drweb32.dll)と、集中管理モードで作業するための設定。
update/	– ダウンロード中に更新を一時的に保存するためのディレクトリ。

ディレクトリで使用される文字・記号の詳細については、はじめにを参照してください。

コンポーネントのカスタムインストールとアンインストール

必要に応じ、該当するそれぞれのパッケージをインストール／アンインストールすることで、特定のDr.Web for UNIX Internet Gatewaysコンポーネントのみをインストール／アンインストールできます。カスタムコンポーネントのインストールまたはアンインストールは、製品のインストールと同じ方法で実行します。

コンポーネントを再インストールするには、まず初めにそのコンポーネントをアンインストールし、その後再度インストールしてください。

- カスタムインストール用の一般的なコンポーネントキット。
- Dr.Web for UNIX Internet Gatewaysコンポーネントのインストールとアンインストール:
  - リポジトリからインストールする;
  - ユニバーサルパッケージからインストールする

カスタムインストール用の一般的なコンポーネントキット

リポジトリまたはユニバーサルパッケージからルートメタパッケージをインストールする代わりに、機能を制限してDr.Web for UNIX Internet Gatewaysをインストールする必要がある場合は、必要な機能を提供するコンポーネントパッケージのみをインストールできます。依存関係を解決するために必要なパッケージは自動的にインストールされます。以下の表は、一般的なDr.Web for UNIX Internet Gatewaysタスクを解決するために設計されたコンポーネントセットを示しています。**インストールするパッケージ**列には、特定のコンポーネントスイートを取得するためにインストールする必要があるパッケージのリストがあります。

カスタムコンポーネントキット	インストールするパッケージ	インストールされるコンポーネント
コンソールスキャンのための最小キット	drweb-filecheck	• Dr.Web Ctl • Dr.Web ConfigD • Dr.Web Scanning Engine • Dr.Web File Checker • Dr.Web Updater • ウイルスデータベース

カスタムコンポーネントキット	インストールするパッケージ	インストールされるコンポーネント
ClamAVのエミュレーションのためのスイート（**clamd**）	`drweb-clamd`	• Dr.Web Ctl • Dr.Web ConfigD • Dr.Web Scanning Engine • Dr.Web File Checker • Dr.Web Network Checker • Dr.Web Updater • Dr.Web ClamD • ウイルスデータベース
ICAPプロトコル経由でのプロキシサーバーを使用したWebサイトへのアクセスをチェックするためのスイート（アンチウイルストラフィックスキャンなし）	`drweb-icapd`	• Dr.Web Ctl • Dr.Web ConfigD • Dr.Web ICAPD • Dr.Web Updater • Webリソースカテゴリーのデータベース
ICAPプロトコル経由でのプロキシサーバーを使用したWebサイトへのアクセスをチェックするためのスイート（アンチウイルストラフィックスキャンあり）。 *注意：Dr.Web Network Checkerを介してスキャン用のデータを受信する別のサーバーでアンチウイルススキャンが実行される場合、`drweb-se`パッケージのインストールがスキップされる可能性があります。* **マークは、`drweb-se`パッケージがインストールされていない場合にインストールされないコンポーネントを示しています。*	`drweb-icapd` `drweb-netcheck` `drweb-se *`	• Dr.Web Ctl • Dr.Web ConfigD • Dr.Web ICAPD • Dr.Web Network Checker • Dr.Web Scanning Engine * • Dr.Web Updater • ウイルスデータベース* • Webリソースカテゴリーのデータベース
HTTP接続のローカルスキャンのためのスイート *注意：接続のアンチウイルススキャンが必要ない場合は、`drweb-netcheck`および`drweb-se`パッケージをインストールする必要はありません。Dr.Web Network Checkerを介してスキャン用のデータを受信する別のサーバーでアンチウイルススキャンが実行される場合、`drweb-se`パッケージのインストールがスキップされる可能性があります。望ましくないWebリソースのカテゴリーにURLが該当するかをチェックする必要がない場合は、`drweb-dws`パッケージのインストールをスキップできます。*	`drweb-gated` `drweb-firewall` `drweb-netcheck *` `drweb-se *` `drweb-dws **`	• Dr.Web Ctl • Dr.Web ConfigD • SpIDer Gate • Dr.Web Firewall for Linux • Dr.Web Network Checker • Dr.Web Scanning Engine * • Dr.Web Updater *** • ウイルスデータベース* • Webリソースカテゴリーのデータベース **

カスタムコンポーネントキット	インストールするパッケージ	インストールされるコンポーネント
*マークは、drweb-seパッケージがインストールされていない場合にインストールされないコンポーネントを示しています。**マークは、drweb-dwsパッケージがインストールされていない場合にインストールされないコンポーネントを示しています。Dr.Web Updaterコンポーネント（***マークが付いています）は、ウイルスデータベースまたはWebリソースカテゴリーのデータベースがインストールされている場合にのみインストールされます。		

リポジトリからインストールされたDr.Web for UNIX Internet Gatewaysコンポーネントのインストールとアンインストール

Dr.Web for UNIX Internet Gatewaysをリポジトリからインストールした場合、コンポーネントのカスタムインストール／アンインストールには、お使いのOSで使用されているパッケージマネージャーのコマンドを使用します。以下はその例です。

1. **CentOS**上にインストールされているDr.Web for UNIX Internet GatewaysからDr.Web ClamD（drweb-clamdパッケージ）をアンインストールするには、次のコマンドを使用します。

```
# yum remove drweb-clamd
```

2. **Ubuntu**にインストールされているDr.Web for UNIX Internet GatewaysにDr.Web ClamD（drweb-clamdパッケージ）を追加でインストールするには、以下のコマンドを使用します。

```
# apt-get install drweb-clamd
```

必要に応じて、お使いのOSで使用されているパッケージマネージャーのヘルプを参照してください。

ユニバーサルパッケージからインストールされたDr.Web for UNIX Internet Gatewaysコンポーネントのインストールとアンインストール

Dr.Web for UNIX Internet Gatewaysがユニバーサルパッケージからインストールされていて、コンポーネントのパッケージを追加でインストールまたは再インストールする場合、Dr.Web for UNIX Internet Gatewaysのインストール元のインストールファイル（.run拡張子の付いたファイル）が必要です。このファイルを保存していない場合は、Doctor WebのWebサイトからダウンロードしてください。

インストールファイルを展開する

.runファイルを実行する際は、以下のコマンドラインパラメータを指定することもできます。

--noexec - インストールプロセスを開始せずに、Dr.Web for UNIX Internet Gatewaysのインストールファイルを展開します。ファイルは TMPDIR 環境変数で指定されたディレクトリに置かれます（通常は /tmp）。

`--keep` - インストール完了後にDr.Web for UNIX Internet Gatewaysのインストールファイルとインストールログを自動的に削除しません。

`--target` *<directory>* - Dr.Web for UNIX Internet Gatewaysのインストールファイルを、指定されたディレクトリ *<directory>* に展開します。

.runファイルの起動時に指定できるコマンドラインパラメータの一覧を表示するには、以下のコマンドを入力します。

```
$ ./<file_name>.run --help
```

カスタムインストールでは、展開されたインストールファイルを使う必要があります。それらのファイルが含まれたディレクトリがない場合、初めに展開します。その際、以下のコマンドを入力します。

```
$ ./<file_name>.run --noexec --target <directory>
```

コマンドが実行された後、ディレクトリ *<directory>* 内に、ネストされたディレクトリの名前 *<file_name>* が現れます。

コンポーネントのカスタムインストール

RUNインストールファイルには、Dr.Web for UNIX Internet Gateways のすべてのコンポーネントのパッケージ（RPMフォーマットで）とサポートファイルが含まれています。各コンポーネントのパッケージファイルは以下の構造を持っています。

```
<component_name>_<version>~linux_<platform>.rpm
```

<version> は製品リリースのバージョンと時間が含まれたストリングで、*<platfrom>* はDr.Web for UNIX Internet Gatewaysが対象としているプラットフォームです。Dr.Web for UNIX Internet Gatewaysのコンポーネントが含まれているパッケージの名前はすべて「drweb」プレフィックスで始まります。

パッケージマネージャーはインストールキットでのパッケージのインストール時に有効になります。カスタムインストールでは、サービススクリプト `installpkg.sh` を使用する必要があります。その際、まずインストールパッケージのコンテンツをディレクトリに展開する必要があります。

パッケージをインストールするには、スーパーユーザー権限（*root* ユーザーの権限）が必要です。権限を昇格するには、**su** コマンド（カレントユーザーを変更する）または **su** コマンド（指定されたコマンドを別のユーザーの権限で実行する）を使用します。

コンポーネントパッケージのインストールまたは再インストールを開始するには、展開されたインストールキットのあるディレクトリに行き、コンソール経由で以下のコマンドを実行します（または、グラフィカルモードのターミナルエミュレーター経由）。

```
# ./scripts/installpkg.sh <package_name>
```

例:

```
# ./scripts/installpkg.sh drweb-clamd
```

Dr.Web for UNIX Internet Gateways全体のインストールを開始する必要がある場合は、以下のコマンドを使用して自動インストールスクリプトを実行してください。

```
$ ./install.sh
```

その他、製品のルートメタパッケージを実行することで、すべてのDr.Web for UNIX Internet Gatewaysパッケージをインストールできます（欠けているか、誤って削除してしまったコンポーネントをインストールするため）。

```
# ./scripts/installpkg.sh drweb-internet-gateways
```

コンポーネントのカスタムアンインストール

お使いのOSがRPMフォーマットのパッケージを使用している場合、コンポーネントのカスタムアンインストールでは、OSのパッケージマネージャーの該当するアンインストールコマンドを使用します。

- **Red Hat Enterprise Linux**と**CentOS**では、**yum** `remove` *<package_name>*コマンドを使用します。
- **Fedora**では、**yum** `remove` *<package_name>*または**dnf** `remove` *<package_name>*コマンドを使用します。
- **SUSE Linux**では、**zypper** `remove` *<package_name>*コマンドを使用します。
- **Mageia**と **OpenMandriva Lx**では、**urpme** *<package_name>*コマンドを使用します。
- **Alt Linux**と**PCLinuxOS**では、**apt-get** `remove` *<package_name>*コマンドを使用します。

例（**Red Hat Enterprise Linux** の場合）:

```
# yum remove drweb-clamd
```

お使いのOSがDEBパッケージを使用している場合（**MSVS** 3.0 OSを使用している場合も）、またはOSにパッケージマネージャーがない場合（**FreeBSD**）、カスタムアンインストールでは、Dr.Web for UNIX Internet Gatewaysのインストール中に自動的にインストールされるパッケージマネージャー**zypper**を使用する必要があります。これを行うには、*<opt_dir>*`/bin`（**GNU/Linux**の場合は`/opt/drweb.com/bin`）ディレクトリに移動して、以下のコマンドを実行します。

```
# ./zypper remove <package_name>
```

例:

```
# ./zypper remove drweb-clamd
```

Dr.Web for UNIX Internet Gateways をアンインストールする必要がある場合は、以下のコマンドを入力して 自動削除 スクリプトを実行します。

```
# ./uninst.sh
```

コンポーネントを再インストールするには、まずそのコンポーネントをアンインストールし、その後、インストールキットからのカスタムインストールまたはフルインストールを実行することで再度インストールします。

セキュリティサブシステムを設定する

OSに強化セキュリティサブシステム**SELinux**が実装されている場合や、**PARSEC**などの強制アクセス制御システム（UNIXで使用されていた従来の任意モデルではなく）が使用されている場合は、それらがデフォルト設定になっているとDr.Web for UNIX Internet Gatewaysとの動作に問題が生じます。この場合、Dr.Web for UNIX Internet Gatewaysが確実に正常に動作するよう、セキュリティサブシステムやDr.Web for UNIX Internet Gatewaysの設定を変更する必要があります。

この章では、**SELinux**セキュリティポリシーの設定の詳細について説明します。

SELinux セキュリティポリシーを設定する

GNU/Linuxディストリビューションに**SELinux**（*Security-Enhanced Linux*）が含まれている場合は、インストール後にDr.Web for UNIX Internet Gatewaysのサービスコンポーネント（スキャンエンジンなど）を正常に動作せるために、**SELinux**のセキュリティポリシーを設定することが必要になる場合があります。

1.ユニバーサルパッケージを使用したインストールの問題

SELinux が有効になっている場合、Dr.Web for UNIX Internet Gatewaysコンポーネントを動作させる *drweb* ユーザーの作成がブロックされることがあり、インストールファイル（`.run`）からのインストールは失敗する場合があります。

*drweb*ユーザーを作成できないためにファイルからのDr.Web for UNIX Internet Gatewaysのインストールが失敗する場合は、**getenforce**コマンドを使用して**SELinux**の動作モードを確認してください。このコマンドは、現在のスキャンモードを出力します。

- `Permissive` - 保護は有効ですが、許可方式が使用されています。セキュリティポリシーに違反する動作は拒否されませんが、動作に関する情報はログに記録されます。
- `Enforced` - 保護は有効で、制御方式が使用されています。セキュリティポリシーに違反する動作は拒否され、動作に関する情報はログに記録されます。
- `Disabled` - **SELinux** はインストールされていますが、有効になっていません。

SELinux が *Enforced* モードで動作している場合は *Permissive* モードに変更してください。その際、以下のコマンドを使用します。

```
# setenforce 0
```

このコマンドは*SELinux*の**Permissive**モードを一時的に（次の再起動まで）有効にします。

> **setenforce** コマンドで有効にした動作モードに関係なく、OSの再起動後、**SELinux** は設定内で指定された動作モードに戻りますので注意してください（**SELinux**の設定ファイルは通常、`/etc/selinux` ディレクトリにあります）。

Dr.Web for UNIX Internet Gatewaysが正常にインストールされた後、製品を起動させる前に *Enforced* モードを再度有効にしてください。その際、以下のコマンドを使用します。

```
# setenforce 1
```

2. Dr.Web for UNIX Internet Gatewaysの動作に関する問題

SELinuxの実行中にいくつかのDr.Web for UNIX Internet Gatewaysコンポーネント（**drweb-se**や**drweb-filecheck**など）が起動できないことがあります。これにより、オブジェクトのスキャンやファイルシステムのモニタリングが不可能になります。これらのコンポーネントが起動できない場合は、**syslog**サービスによって管理されるシステムログ（通常このログは/var/log/ディレクトリにあります）に*119*および*120*エラーメッセージが表示されます。

SELinux セキュリティシステムによってアクセスが拒否された場合、そのようなイベントのログが記録されます。一般的に、システムで **audit** デーモンが使用されている場合、audit（監査）に関するログが /var/log/audit/audit.log ファイルに保存されます。それ以外の場合、ブロックされた動作に関するメッセージが一般的なログファイル（/var/log/messages または /var/log/syslog）に保存されます。

SELinuxにブロックされているために製品のスキャンコンポーネントが機能しない場合は、コンポーネント用の特別な*セキュリティポリシー*を設定する必要があります。

> 一部の**GNU/Linux**ディストリビューションには、下記のユーティリティがありません。その場合は、これらのユーティリティを備えた追加パッケージのインストールが必要になる場合があります。

SELinux セキュリティポリシーを設定するには以下の手順を行ってください。

1. **SELinux** のポリシーソースコードの新しいファイルを作成します（.te ファイル）。このファイルは記載されているポリシーモジュールに関連した制限を規定するものです。このポリシーソースコードは以下のいずれかの方法で作成できます。
   1) **audit2allow** ユーティリティを使用して - 最もシンプルな方法です。ユーティリティはシステムログファイル内のアクセス拒否に関するメッセージからpermissiveルールを生成します。自動でメッセージを検索するよう設定するか、手動でログファイルへのパスを指定できます。

      この方法は、Dr.Web for UNIX Internet Gatewaysのコンポーネントが **SELinux** のセキュリティポリシーに違反していて、それらのイベントが監査ログファイルに記録されている場合のみ使用できます。そうでない場合、そのようなイベントが起こるのを待つか、**policygentool** ユーティリティを使用して強制的にpermissiveポリシーを作成（下記参照）してください。

      > **audit2allow** ユーティリティは policycoreutils-python パッケージ、policycoreutils-devel パッケージ（**RedHat Enterprise Linux**、**CentOS**、**Fedora**、バージョンによる）、またはpython-sepolgen パッケージ（**Debian**、**Ubuntu**）のいずれかにあります。

      audit2allow の使用例：

      ```
      # grep drweb-se.real /var/log/audit/audit.log | audit2allow -M drweb-se
      ```

      この例では、**drweb-se**コンポーネントに対するアクセス拒否メッセージを見つけるために、**audit2allow**ユーティリティが/var/log/audit/audit.logファイル内で検索を実行します。

      ポリシーソースファイルdrweb-se.teと、インストール可能なdrweb-se.ppポリシーモジュールの2つのファイルが作成されます。

システム監査ログ内でセキュリティ違反イベントが見つからなかった場合、ユーティリティはエラーメッセージを返します。

ほとんどの場合、**audit2allow**ユーティリティによって作成されたポリシーファイルを変更する必要はありません。したがって、手順4の`drweb-se.pp`ポリシーモジュールのインストールに進むことを推奨します。**audit2allow**ユーティリティは**semodule**コマンドの呼び出しを出力します。出力をコマンドラインにコピーして実行すると、手順4が完了します。Dr.Web for UNIX Internet Gatewaysコンポーネント用に自動的に生成されたセキュリティポリシーを変更する場合のみ、手順2に進みます。

2) **policygentool**ユーティリティを使用する。そのためには、異なる方法で処理するコンポーネントの名前とその実行ファイルへのフルパスを指定します。

> **RedHat Enterprise Linux** と **CentOS Linux** 向けの `selinux-policy` パッケージに含まれている **policygentool** ユーティリティは正常に機能しない場合があります。その場合は **audit2allow** ユーティリティを使用してください。

policygentool を使用したポリシー作成の例：

- **drweb-se**コンポーネントの場合：

```
# policygentool drweb-se /opt/drweb.com/bin/drweb-se.real
```

- **drweb-filecheck**コンポーネントの場合：

```
# policygentool drweb-filecheck /opt/drweb.com/bin/drweb-filecheck.real
```

ドメインを作成するための一般的なプロパティをいくつか指定するように求められます。その後、ポリシーを決定する以下の3つのファイルが（コンポーネントごとに）作成されます。

<module_name>.te、*<module_name>.fc*、*<module_name>.if*。

2. 必要に応じ、生成されたポリシーソースファイル *<module_name>.te* を編集し、その後、**checkmodule** ユーティリティを使用して、ローカルポリシーのこのソースファイルをバイナリ形式に変換（`.mod` ファイル）します。

> コマンドを正常に実行するには、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。

使用例：

```
# checkmodule -M -m -o drweb-se.mod drweb-se.te
```

3. **semodule_package** ユーティリティを使用して、インストール用のポリシーモジュールを作成します（`.pp` ファイル）。

例：

```
# semodule_package -o drweb-se.pp -m drweb-se.mod
```

4. 作成されたポリシーモジュールをインストールするには、**semodule** ユーティリティを使用します。

例：

```
# semodule -i drweb-se.pp
```

SELinuxの動作と設定に関する詳細は、お使いの**Linux**ディストリビューションのマニュアルを参照してください。

開始する

1. インストールされたDr.Web for UNIX Internet Gatewaysの使用を開始するために、キーファイルを入手してインストールした上で有効化する必要があります。
2. Dr.Web for UNIX Internet Gatewaysの操作性をさらにスキャンすることをお勧めします。
3. Dr.Web for UNIX Internet Gatewaysを、使用しているHTTPプロキシサーバーと統合します（提供されている**Squid**プロキシサーバーとの統合に関する説明を参照してください）。
4. ローカルWebサーバーを外部ネットワークの脅威から保護するために、SpIDer Gateモニターの設定を変更します。
5. 外部のプロキシサーバーを使用していない場合は、SpIDer Gateのプロキシモードを設定します。
6. サーバーの保護に必要な場合には、どのコンポーネントが実行されているかを確認し、追加のコンポーネントを有効にします（ディストリビューションに依存しますが、Dr.Web ClamD、Dr.Web SNMPD等）。追加コンポーネントを有効にすること以外に、他のアクションを実行する必要がある場合があります。たとえば、デフォルトの設定を調整する必要があります。インストール済みおよび実行中のコンポーネントとその設定の一覧を表示するには、次のいずれかを使用します。
   - コマンドラインベースの管理ツール-Dr.Web Ctl（**drweb-ctl** `appinfo`、**drweb-ctl** `cfshow`、**drweb-ctl** `cfset`コマンドを使用します）。
   - Dr.Web for UNIX Internet Gatewaysの管理用ウェブインターフェース（初期設定では、ウェブブラウザから`https://127.0.0.1:4443/`にアクセスすると利用できます）。

製品の登録と有効化

このセクションの内容：

ライセンスを購入・登録する

ライセンスを購入すると、製品コンポーネントとウイルスデータベースの更新がDoctor Web更新サーバーから定期的にダウンロードされます。さらに、購入した製品をインストールまたは使用するときに問題が発生した場合は、Doctor Webまたはそのパートナーが提供するテクニカルサポートサービスを利用できます。

Dr.Web製品の購入や製品のシリアル番号の入手は、パートナーまたはオンラインストアから可能です。ライセンスオプションの詳細については、Doctor Web公式Webサイト（https://license.drweb.com/）にアクセスしてください。

ライセンス登録は、ユーザーがDr.Web for UNIX Internet Gatewaysの正規ユーザーであることを証明し、ウイルスデータベースの定期的な更新を含むアンチウイルスの機能を有効にするために必要です。インストールが完了したら、製品を登録してライセンスを有効化することを推奨します。購入したライセンスは、Doctor Webの公式Webサイト（https://products.drweb.com/register/）で有効化できます。

有効化の際には、購入したライセンスのシリアル番号を入力する必要があります。シリアル番号はDr.Web for UNIX Internet Gatewaysと一緒に提供されるか、オンラインでライセンスを購入または更新した際にメールで提供されます。

これまでに製品を使用したことがある場合は、新しいライセンスに対して150日間の延長ボーナスを利用できます。このボーナスを有効にするには、登録済みのシリアル番号を入力するか、ライセンスキーファイルを提供します。

複数のサーバー上で Dr.Web for UNIX Internet Gateways を使用するための複数のライセンスがあり、製品を1台のサーバーでのみ使用することを選択した場合、それを指定することができ、ライセンス有効期限は自動的に延長されます。

デモライセンスを取得する

ご利用のDr.Web for UNIX Internet Gatewaysの試用期間は、Doctor Webの公式Webサイト（https://download.drweb.com/demoreq/biz/）で確認できます。製品を選択して登録フォームに記入すると、Dr.Web for UNIX Internet Gatewaysを有効化するためのシリアル番号またはキーファイルが記載されたメールが届きます。

同じコンピューターでの2回目以降の試用期間は、一定の期間が経過した後に利用できます。

Dr.Web Ctl（**drweb-ctl**）コマンドラインツールのライセンスコマンドを使用すると、登録されたライセンスのシリアル番号のデモキーファイルまたはライセンスキーファイルを自動的に取得できます。

キーファイルのインストール

キーファイルは、Dr.Web for UNIX Internet Gateways の購入したライセンスまたは有効化した試用期間に対応する、ローカルコンピューター上に保存される特別なファイルです。このファイルには提供されたライセンスまたは試用期間に関する情報が含まれ、また、このファイルに応じて使用権が規定されます。

Dr.Web for UNIX Internet Gatewaysの動作中、キーファイルはデフォルトの *<etc_dir>*（**GNU/Linux**の場合は `/etc/opt/drweb.com`）ディレクトリ内に`drweb32.key`という名前で置かれている必要があります。

Dr.Web for UNIX Internet Gatewaysのコンポーネントは、キーファイルが使用可能かつ有効であるかどうかを定期的に確認します。改竄されることを防ぐため、キーファイルはデジタル署名されています。キーファイルを編集すると無効になります。誤って無効にしてしまうことを防ぐため、キーファイルをテキストエディターで開かないようにすることが推奨されます。

有効なキーファイル（正規またはデモライセンス）が見つからない場合、またはライセンスの有効期限が切れている場合、有効なキーファイルがインストールされるまでアンチウイルスコンポーネントの動作はブロックされます。

ライセンスキーファイルは有効期限が切れるまで保管しておき、Dr.Web for UNIX Internet Gatewaysの再インストールや、別のコンピューターへのインストールにはそのキーを使用することを推奨します。この場合、登録時に指定したものと同じ製品シリアル番号と顧客データを使用する必要があります。

メールメッセージでは、Dr.Webキーファイルは通常、zipアーカイブに圧縮されて転送されます。Dr.Web for UNIX Internet Gatewaysアクティベーション用のキーファイルを含むアーカイブは、通常は`agent.zip`という名前です（メッセージに*複数の*アーカイブが含まれている場合は、`agent.zip`アーカイブを使用する必要があります）。キーファイルをインストールする前に、アーカイブを適宜解凍し、そこからキーファイルを抽出して、使用可能な任意のディレクトリ（たとえば、ホームディレクトリやUSBフラッシュドライブ）に保存してください。

製品の有効なライセンスに対応するキーファイルをお持ちの場合（キーファイルをメールで受け取った場合、またはDr.Web for UNIX Internet Gatewaysを別のサーバー上で使用する場合など）、そのキーファイルへのパスを指定することでDr.Web for UNIX Internet Gatewaysを有効化できます。その場合は、次の操作を行います。

1. アーカイブの場合はキーファイルを展開します。
2. 次のいずれかを実行してください:
   - キーファイルを*<etc_dir>*ディレクトリにコピーし、必要に応じてファイル名を`drweb32.key`に変更します。
   - Dr.Web for UNIX Internet Gateways設定ファイルで、**KeyPath**パラメータ値にキーファイルパスを指定します。
3. 次のコマンドを入力して、Dr.Web for UNIX Internet Gatewaysの設定をリロードします。

   ```
   # drweb-ctl reload
   ```

   すべての変更が適用されます。

また、次の コマンド を使用することもできます。

```
# drweb-ctl cfset Root.KeyPath <path to the key file>
```

この場合、Dr.Web for UNIX Internet Gatewaysの再起動は不要です。キーファイルは*<etc_dir>*ディレクトリにコピーされず、元の場所に残ります。

<opt_dir>、*<etc_dir>*、*<var_dir>*の表記規則の詳細は、はじめにを参照してください。

キーファイルが*<etc_dir>*ディレクトリにコピーされない場合は、ユーザーはファイルが破損や削除から保護されていることを確認する必要があります。キーファイルがシステムから誤って削除される可能性があるため、この方法は推奨されません（たとえば、キーファイルが存在するディレクトリが定期的にクリーンアップされる場合）。キーファイルを紛失した場合は、サポートを要求して新しいファイルを取得できますが、要求できる回数には制限があります。

2回目以降の登録

キーファイルを紛失したが、既存のライセンスの有効期限が切れていない場合は、前回の登録時に指定した個人データを入力して、再度登録する必要があります。別のメールアドレスを使用できます。この場合、ライセンスキーファイルは新しく指定されたアドレスに送信されます。

キーファイルをリクエストできる回数は制限されています。1つのシリアル番号は*最大25回まで*登録できます。その数を超えてリクエストが送信された場合、キーファイルは配信されません。紛失したキーファイルを入手するには、

Doctor Webテクニカルサポートに連絡して問題の詳細を説明し、シリアル番号の登録時に入力した個人データを伝えてください。ライセンスキーファイルはメールで送信されます。

キーファイルがメールで送信されたら、手動でインストールする必要があります。

製品の動作確認

*EICAR (European Institute for Computer Anti-Virus Research)*テストは、ウイルスをシグネチャで検出するアンチウイルスプログラムの動作を確認できます。このテストは、インストールされたアンチウイルスツールのウイルス検出の動作を、コンピューターを危険にさらすことなくテストするために特別に設計されています。

*EICAR*テストは実際にはウイルスではありませんが、大多数のアンチウイルスによってウイルスのように扱われます。この「ウイルス」を検出すると、Dr.Webアンチウイルス製品は**EICAR Test File (NOT a Virus!)**を報告します。他のアンチウイルスツールも同様にユーザーに警告します。**EICAR**テストファイルは、実行時に端末画面またはコンソールエミュレーターに次の行を出力する**MS DOS**/**MS Windows**用の68バイトのCOMファイルです。

```
EICAR-STANDARD-ANTIVIRUS-TEST-FILE!
```

EICAR テストファイルは、次の文字列のみを含んでいます。

```
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

上の文字列でファイルを作成すると、「ウイルス」として認識されるテストファイルができあがります。

Dr.Web for UNIX Internet Gatewaysが正常に動作していれば、テストファイルはスキャンの種類に関係なくファイルシステムのスキャン中に検出され、検出された脅威についてユーザーに対して通知が行われます：
EICAR Test File (NOT a Virus!)

コマンドラインから**EICAR**テストを使用してDr.Web for UNIX Internet Gatewaysの動作を確認するコマンドの例：

```
$ tail <opt_dir>/share/doc/drweb-se/readme.eicar | grep X5O > testfile &&
drweb-ctl rawscan testfile && rm testfile
```

このコマンドは、*<opt_dir>*`/share/doc/drweb-se/readme.eicar`ファイル（Dr.Web for UNIX Internet Gatewaysに付属）から**EICAR**テストファイルの本文を表す文字列を抽出し、それを現在のディレクトリに作成された`testfile`という名前のファイルに書き込みます。次にこのファイルをスキャンし、その後でファイルを削除します。

上記のテストを行うには、カレントディレクトリへの書き込みアクセスが必要です。また、ディレクトリに `testfile` という名前のファイルが含まれていないことを確認してください（必要に応じ、コマンド内でファイル名を変更してください）。

<opt_dir>、*<etc_dir>*、*<var_dir>*の表記規則の詳細は、はじめにを参照してください。

テストウイルスが検出されると、以下のメッセージが表示されます。

```
<path to the current directory>/testfile - infected with EICAR Test File (NOT a Virus!)
```

テスト中にエラーが発生した場合は、既知のエラーを参照してください（付録F. 既知のエラーを参照）。

> SpIDer Guardが有効になっている場合、悪意のあるファイルはただちに削除または隔離できます（コンポーネントの設定によって異なります）。この場合、コマンド**rm**はファイルが見つからないことを通知します。これはモニターが通常モードで動作していることを意味します。

受信HTTPトラフィックにウイルスが含まれているかどうかをテストするには：

Webブラウザ経由

1. ブラウザを開き、プロキシサーバー設定に移動します。
2. ICAPの適切なプロキシサーバー設定を入力します。
3. Webページhttp://2016.eicar.org/download/eicar.comにアクセスします。ファイルが感染しているという通知がブラウザウィンドウに表示されます。

コンソール経由

次のリクエストを行います。

```
curl -x 127.0.0.1:3128 http://2016.eicar.org/download/eicar.com
```

リクエストしたファイルが感染しているという通知が応答に表示されます。

Squidプロキシサーバーとの統合

このセクションの内容：

- Dr.Web ICAPDを設定する
- Squidを設定する
- Squidの詳細設定

1）Dr.Web ICAPDを設定する

Dr.Web ICAPDと**Squid** HTTPプロキシサーバーを統合するには、設定ファイル内のDr.Web ICAPDの設定セクション（`[ICAPD]`セクション）で現在のパラメータ値を確認し、必要に応じて変更する必要があります。

- **`ListenAddress`**パラメータに、HTTPプロキシサーバーからの接続を待機するDr.Web ICAPDによって待ち受け（リッスン）されるネットワークソケットのアドレス（*<IP address>:<port>*）を指定します（デフォルトでは`127.0.0.1:1344`ソケットを使用）。
- **`Block*`**設定で、Dr.Web ICAPDがブロックまたは許可するWebサイトのカテゴリーと脅威の種類を有効または無効にします。
- 必要に応じて、**`WhiteList`**および**`BlackList`**のパラメータを使用して、ブロックしないWebサイトとブロックするWebサイトを定義できます。**`BlackList`**パラメータは**`WhiteList`**パラメータよりも優先されます。つまり、両方のパラメータの値に同じWebサイトが含まれている場合、このWebサイトへのアクセスはブロックされます。

- Webサイトへのアクセスをより詳細に（さまざまな条件に基づいて）設定するには、スキャンルールを編集します。

> 設定のDr.Web ICAPDのセクションにある**UsePreview**、**Use204**、**AllowEarlyResponse**パラメータのデフォルト値を使用すると、Internet Content Adaptation Protocol（ICAP）の該当する機能をコンポーネントで使用できます（つまり、*ICAPプレビュー*モードを使用する、*ICAPプレビュー*モード以外で204ステータスコードを返す、要求全体がプロキシサーバーから受信される前に「初期」応答の送信を開始するといった操作が可能になります）。HTTP要求処理に問題がない場合は、デフォルト値を変更しないことをお勧めします。

すべての設定を調整したら、Dr.Web for UNIX Internet Gatewaysを再起動します（コマンド**drweb-ctl** `reload`を使用します）。設定デーモンDr.Web ConfigDを再起動することもできます（**service** `drweb-configd restart`コマンドを使用します）。

2）Squidを設定する

SquidとDr.Web ICAPDとのインタラクションを有効にするには、`squid.conf`設定ファイル（通常は`/etc/squid3/`にあります）を編集してICAPの使用を許可します。**Squid**を設定するには、次のパラメータを設定します。

1. **Squid**にICAPの使用を許可する。
2. **Squid**が使用するICAPサービスとしてDr.Web ICAPDを登録する。
3. *ICAPプレビュー*モード（オプション）の使用を有効にする。
4. クライアントのデータ（プロキシサーバーでの認証にパスしたユーザーのIPアドレスとユーザー名）を送信し、Dr.Web ICAPDのルール内で使用することを許可する（オプション）。
5. Dr.Web ICAPDと**Squid**間の常時接続のサポートを有効にする（オプション。常時接続の使用は必須ではありませんが、**Squid**とDr.Web ICAPDを同時に使用する際のパフォーマンスが向上します）。

Squidを設定するときは、次の点に注意してください。

- **Squid**でICAP経由のHTTP要求（*REQMOD*）とHTTP応答（*RESPMOD*）をチェックするには、該当する種類の2つのICAPサービスを追加します。
- **Squid**でDr.Web ICAPDをICAPサービスとして使用するには、`icap_service`で指定したアドレスとポートが、Dr.Web ICAPDの設定の**ListenAddress**パラメータで指定したアドレスとポートと一致する必要があります。
- **icap_preview_size**パラメータ値が**0**でない場合、Dr.Web ICAPDは`Squid`では機能しません。
- **Squid**は自動的に「クライアントのIPアドレス」と「ユーザー名」の値を作成し、ICAP要求のヘッダーとしてDr.Web ICAPDにリダイレクトします。このデータの正確性と可用性は保証されていません。Dr.Web ICAPDは、ユーザー名とユーザーのIPアドレスがプロキシサーバーによって`X-Client-Username`ヘッダーと`X-Client-IP`ヘッダーで転送されると想定します。また、**Squid**の設定でデフォルトで定義されている値エンコード方法のみが使用されると想定します。このため、**Squid**を設定するときは、このデータの転送方法に影響を与えるパラメータ値（`icap_client_username_encode`や`icap_client_username_header`など）を変更しないことをお勧めします。

> 使用する**Squid**バージョンはICAPを有効化（`--enable-icap-client`オプションを指定してコンパイル）して構築される必要があります。ICAPが有効化されていない場合は、**Squid**とDr.Web ICAPDの間に接続を確立することができません。

設定できるパラメータのリストは、使用している**Squid**サーバーのバージョンによって異なります（**Squid**バージョン3.2（およびそれ以降）、3.1、3.0の設定については、後述の説明を参照）。下の文字列がすでに設定ファイルにある場合は、それらの値を指定した値に変更する必要があります。該当のパラメータがすでにファイルに含まれていても、コメントアウトされている場合は、コメントを外します。**Squid**設定ファイルに必要なパラメータがない場合は、それらのパラメータをファイルに追加します（末尾など）。

Dr.Web ICAPDと**Squid**の間のやり取りを設定するには、#1と#2の手順のみが必須です。以下のうち、他の設定が不要な場合は、それらを**Squid**設定ファイルに追加しないでください。

Squid 3.2以降のバージョンの場合

```
#1
icap_enable on

#2
icap_service i_req reqmod_precache bypass=0 icap://127.0.0.1:1344/reqmod
icap_service i_res respmod_precache bypass=0 icap://127.0.0.1:1344/respmod

adaptation_access i_req allow all
adaptation_access i_res allow all

#3
icap_preview_enable on
icap_preview_size 0

#4（Squid 3.2では、icap_send_client_ipと
#icap_send_client_usernameパラメータの名前が変更されました）
adaptation_send_client_ip on
adaptation_send_username on

#5
icap_persistent_connections on
```

Squid 3.1の場合

```
#1
icap_enable on

#2（Squid 3.1では、サービスの設定に使用される形式が変更され
# icap_accessパラメータの名前が変更されました）
icap_service i_req reqmod_precache bypass=0 icap://127.0.0.1:1344/reqmod
icap_service i_res respmod_precache bypass=0 icap://127.0.0.1:1344/respmod

adaptation_access i_req allow all
adaptation_access i_res allow all

#3
icap_preview_enable on
icap_preview_size 0
```

```
#4
icap_send_client_ip on
icap_send_client_username on

#5
icap_persistent_connections on
```

Squid 3.0の場合

```
#1
icap_enable on

#2
icap_service i_req reqmod_precache 0 icap://127.0.0.1:1344/reqmod
icap_service i_res respmod_precache 0 icap://127.0.0.1:1344/respmod

icap_class icapd_class_req i_req
icap_class icapd_class_resp i_res

icap_access icapd_class_req allow all
icap_access icapd_class_resp allow all

#3
icap_preview_enable on
icap_preview_size 0

#4
icap_send_client_ip on
icap_send_client_username on

#5
icap_persistent_connections on
```

Squidの設定を変更したら、再起動してください。

Squidの詳細設定

必要に応じて、**Squid**がICAPプロトコル経由でスキャンのために送信するデータのサイズを制限できます。この目的のために、設定ファイルはヘッダーContent-Lengthのコンテンツの要件を満たす（あるいは満たさない）条件で追加される必要があります。例：

```
acl <name> rep_header Content-Length ^[0-9]{7,}$
```

（サーバー応答のヘッダーContent-Lengthに999999より大きい数値が含まれている場合、条件*<name>*はtrueになります）。

次に、追加した条件を使用して、ICAPプロトコルを介したサーバー応答のスキャンを許可または拒否します（**Squid**の外部ICAPサーバーへの接続パラメータでは、すべての語を条件名 *<name>*に置き換えます）。上の例は、ヘッダーContent-Lengthが999999よりも大きい数値である場合に当てはまる可能性があるため、条件 *<name>*がtrueである応答のスキャンを*拒否*するために使用します。

```
#Squid 3.1以降のバージョン
adaptation_access i_res deny <name>

#Squid 3.0以降のバージョン
icap_access icapd_class_resp deny <name>
```

ヘッダー Content-Length の存在は、Webサーバーの応答では保証されません。それが利用できない場合、ICAPサーバーにスキャンするために**Squid**から送られるデータのサイズ制限に示された方法は機能しません。

Squidの設定を変更したら、再起動してください。

よりきめ細かい方法でWebトラフィックのスキャンを制限する**Squid**の設定に関する詳細は、**Squid**のドキュメントを参照してください。たとえば、http://www.squid-cache.org/Doc/を参照してください。

ローカルWebサーバーを保護する

このオプションは、**GNU/Linux** OSの製品ディストリビューションでのみ使用できます。

Dr.Web for UNIX Internet Gatewaysがインストールされているのと同じホスト上で実行されているWebサーバーを保護するには、Dr.Web Firewall for Linuxコンポーネントを設定して、WebサーバーへのトラフィックがSpIDer Gateモニターでスキャンされるようにする必要があります。

このセクションの内容：

- 接続のリダイレクトを設定する
- スキャン設定

接続のリダイレクトを設定する

Webサーバー保護を構成するには、Dr.Web Firewall for Linuxの設定のセクション（セクション[LinuxFirewall]）で、設定ファイルのいくつかのパラメータ値を変更します。

パラメータ	必要な値
InspectHttp	On
AutoconfigureIptables	Yes
AutoconfigureRouting	Yes
LocalDeliveryMark	Auto

パラメータ	必要な値
ClientPacketsMark	Auto
ServerPacketsMark	Auto
TproxyListenAddress	127.0.0.1:0 *Dr.Web Firewall for Linuxの操作に特別なIPアドレスまたはポートを使用する場合は、ここで指定します*
InputDivertEnable	Yes
InputDivertNfqueueNumber	Auto
InputDivertConnectTransparently	Yes

Dr.Web Firewall for Linux>の設定を表示および変更するには、次の方法を使用します。

- コマンドラインベースの管理ツール - Dr.Web Ctl(**drweb-ctl** cfshowおよび**drweb-ctl** cfsetコマンドを使用します)。
- Dr.Web for UNIX Internet Gatewaysの管理Webインターフェース(デフォルトでは、Webブラウザから https://127.0.0.1:4443/にアクセスすると利用できます)。

次は、コマンドの例です。

```
# drweb-ctl cfset LinuxFirewall.InputDivertEnable Yes
```

Dr.Web Firewall for Linuxは次のように設定されます。受信接続を介して転送されたデータは、HTTPプロトコルが使用され、対応する**InspectHttp**パラメータ値がOnに設定されている場合に、SpIDer Gateによってスキャンされます。

HTTPSプロトコルを介して転送されたデータをスキャンするには、さらに次の手順を実行します。

- 次のコマンドを実行して対応するパラメータの値を指定することで、SSL/TLS経由で送信されるトラフィックのスキャンを有効にします。

  ```
  # drweb-ctl cfset LinuxFirewall.UnwrapSsl Yes
  ```

  drweb-ctl *ツールのcfsetコマンドまたは管理Webインターフェースを使用することを推奨します。これらを使用した場合、スキャンルールが自動的に更新されるためです。スキャンルールはこのパラメータに依存します。*
- 次のコマンドを実行して、Dr.Web for UNIX Internet Gatewaysが保護されたSSL/TLSチャネルに統合するために使用する証明書をエクスポートします(証明書をPEM形式で保存するために使用されるファイルの名前を指定する必要があります)。

  ```
  $ drweb-ctl certificate > <cert_name>.pem
  ```
- 取得した証明書を信頼できる証明書のシステムリストに追加し、可能な場合には、それをWebクライアント(ブラウザ)およびWebサーバー用の信頼できる証明書として書き込みます。詳細は、付録E. SSL証明書を生成するセクションを参照してください。

スキャン設定

設定ファイルのDr.Web Firewall for Linuxの設定のセクション（[LinuxFirewall]セクション）で次のパラメータを指定する必要があります。

1. 転送データのスキャンパラメータ（**ScanTimeout**、**HeuristicAnalysis**、**PackerMaxLevel**、**ArchiveMaxLevel**、**MailMaxLevel**、**ContainerMaxLevel**、**MaxCompressionRatio**）。これらのパラメータは、スキャンの長さとリソース強度を制限します。きめ細かい設定が不要な場合は、パラメータデータの値をデフォルトの状態にしておくことを推奨します。
2. 不要なURLとコンテンツのブロックパラメータ。そのためには、対応する**Block***パラメータの値を設定します。
3. 受信データをスキャンできない場合のSpIDer Gateの対応を定義する**BlockUnchecked**パラメータ値を設定します。
4. HTTPのフィルタリングルールをよりきめ細かく（さまざまな条件に基づいて）設定するには、Lua procedureまたは**RuleSet**ルールを編集します。

すべての設定を調整したら、Dr.Web for UNIX Internet Gatewaysを再起動します（コマンド**drweb-ctl** `reload`を使用します）。設定デーモンDr.Web ConfigDを再起動することもできます（**service** `drweb-configd restart`コマンドを使用します）。

SpIDer Gateをプロキシモードで使用する

このオプションは、**GNU/Linux** OSの製品ディストリビューションでのみ使用できます。

インターネット経由で広がる脅威からローカルネットワークを保護するため、ICAPまたはClamAVプロトコル経由外部アプリケーションとの統合で（Dr.Web ClamDコンポーネントを直接使用して）Dr.Web for UNIX Internet Gatewaysと通信できるHTTPプロキシサーバーがインターネットゲートウェイにない場合は、Dr.Web for UNIX Internet Gatewaysがインストールされているインターネットゲートウェイ経由で受信した情報がSpIDer Gateモニターによってスキャンされるように Dr.Web Firewall for Linuxを設定します（透過プロキシモード）。

このセクションの内容：

- プロキシモードを設定する
- スキャン設定

プロキシモードを設定する

透過プロキシモードを設定するには、Dr.Web Firewall for Linux（セクション[LinuxFirewall]）の設定のセクションで、設定ファイルの一部のパラメータ値を次のように変更します。

パラメータ	必要な値
InspectHttp	On
AutoconfigureIptables	Yes

パラメータ	必要な値
AutoconfigureRouting	Yes
LocalDeliveryMark	Auto
ClientPacketsMark	Auto
ServerPacketsMark	Auto
TproxyListenAddress	127.0.0.1:0 *Dr.Web Firewall for Linuxの操作に特別なIPアドレスまたはポートを使用する場合は、ここで指定します*
ForwardDivertEnable	Yes
FrowardDivertNfqueueNumber	Auto
ForwardDivertConnectTransparently	Yes

Dr.Web Firewall for Linux>の設定を表示および変更するには、次の方法を使用します。

- コマンドラインベースの管理ツール - Dr.Web Ctl(**drweb-ctl** `cfshow`および**drweb-ctl** `cfset`コマンドを使用します)。
- Dr.Web for UNIX Internet Gatewaysの管理Webインターフェース(デフォルトでは、Webブラウザから`https://127.0.0.1:4443/`にアクセスすると利用できます)。

次は、コマンドの例です。

```
# drweb-ctl cfset LinuxFirewall.ForwardDivertEnable Yes
```

Dr.Web Firewall for Linuxは次のように設定されます。受信接続を介して転送されたデータは、HTTPプロトコルが使用され、対応する**InspectHttp**パラメータ値が`On`に設定されている場合に、SpIDer Gateによってスキャンされます。

HTTPSプロトコルを介して転送されたデータをスキャンするには、さらに次の手順を実行します。

- 次のコマンドを実行して対応するパラメータの値を指定することで、SSL/TLS経由で送信されるトラフィックのスキャンを有効にします。

  ```
  # drweb-ctl cfset LinuxFirewall.UnwrapSsl Yes
  ```

  drweb-ctl *ツールの*`cfset`*コマンドまたは管理Webインターフェースを使用することを推奨します。これらを使用した場合、スキャンルールが自動的に更新されるためです。スキャンルールはこのパラメータに依存します。*
- 次のコマンドを実行して、Dr.Web for UNIX Internet Gatewaysが保護されたSSL/TLSチャネルに統合するために使用する証明書をエクスポートします(証明書をPEM形式で保存するために使用されるファイルの名前を指定する必要があります)。

  ```
  $ drweb-ctl certificate > <cert_name>.pem
  ```

- 取得した証明書を信頼できる証明書のシステムリストに追加し、可能な場合には、それをWebクライアント（ブラウザ）およびWebサーバー用の信頼できる証明書として書き込みます。詳細は、付録E. SSL証明書を生成するセクションを参照してください。

スキャン設定

設定ファイルのDr.Web Firewall for Linuxの設定のセクション（`[LinuxFirewall]`セクション）で次のパラメータを指定する必要があります。

1. 転送データのスキャンパラメータ（**`ScanTimeout`**、**`HeuristicAnalysis`**、**`PackerMaxLevel`**、**`ArchiveMaxLevel`**、**`MailMaxLevel`**、**`ContainerMaxLevel`**、**`MaxCompressionRatio`**）。これらのパラメータは、スキャンの長さとリソース強度を制限します。きめ細かい設定が不要な場合は、パラメータデータの値をデフォルトの状態にしておくことを推奨します。
2. 不要なURLとコンテンツのブロックパラメータ。そのためには、対応する**`Block*`**パラメータの値を設定します。
3. 受信データをスキャンできない場合のSpIDer Gateの対応を定義する**`BlockUnchecked`**パラメータ値を設定します。
4. HTTPのフィルタリングルールをよりきめ細かく（さまざまな条件に基づいて）設定するには、Lua procedureまたは**`RuleSet`**ルールを編集します。

すべての設定を調整したら、Dr.Web for UNIX Internet Gatewaysを再起動します（コマンド**drweb-ctl** `reload`を使用します）。設定デーモンDr.Web ConfigDを再起動することもできます（**service** `drweb-configd restart`コマンドを使用します）。

簡単な説明

このセクションの内容：

- HTTPプロキシとWebサーバーの操作：
  - Dr.Web for UNIX Internet GatewaysをSquidに接続する方法。
  - Webサーバーを保護する方法。
  - SpIDer Gateのプロキシモードを設定する方法。
- Dr.Web for UNIX Internet Gatewaysの一般的な動作：
  - Dr.Web for UNIX Internet Gatewaysを再起動する方法。
  - 集中管理サーバーに接続する方法
  - 集中管理サーバーから切断する方法
  - Dr.Web for UNIX Internet Gatewaysを有効化する方法
  - Dr.Web for UNIX Internet Gatewaysをアップグレードする方法
  - Dr.Web for UNIX Internet Gatewaysコンポーネントを追加または削除する方法
  - Dr.Web for UNIX Internet Gatewaysコンポーネントの動作を管理する方法
  - 製品のログを表示する方法

Dr.Web for UNIX Internet GatewaysをSquidに接続する方法

Squidプロキシサーバーとの統合セクションの指示に従ってください。

Webサーバーを保護する方法

ローカルWebサーバーを保護するセクションの指示に従ってください。

SpIDer Gateのプロキシモードを設定する方法

SpIDer Gateをプロキシモードで使用するセクションの指示に従ってください。

Dr.Web for UNIX Internet Gatewaysを再起動する方法

Dr.Web for UNIX Internet Gatewaysがすでに実行されているときに再起動するには、Dr.Web ConfigD設定デーモンを管理するスクリプトを使用することもできます。デーモンの起動、停止、または再起動は、それぞれDr.Web for UNIX Internet Gatewaysの起動、停止、または再起動を引き起こします。

Dr.Web ConfigDの動作を制御するシェルスクリプトは、標準のOSディレクトリ(**GNU/Linux**の場合は`/etc/init.d/`、**FreeBSD**の場合は`/usr/local/etc/rc.d/`)にあります。スクリプトの名前は`drweb-configd`です。次のパラメータがあります。

パラメータ	説明
`start`	実行していない場合は、Dr.Web ConfigDを起動するように指示します。Dr.Web ConfigDが起動すると、Dr.Web ConfigDはDr.Web for UNIX Internet Gatewaysに必要なすべてのコンポーネントを起動します。
`stop`	Dr.Web ConfigDが実行中の場合はシャットダウンするように指示します。Dr.Web ConfigDがシャットダウンすると、Dr.Web ConfigDはDr.Web for UNIX Internet Gatewaysのすべてのコンポーネントもシャットダウンします。
`restart`	Dr.Web ConfigDを再起動(シャットダウンしてから起動)するように指示します。Dr.Web ConfigDはシャットダウンしてからDr.Web for UNIX Internet Gatewaysのすべてのコンポーネントを起動します。Dr.Web ConfigDが開始されていない場合、パラメータは開始と同じ効果があります。
`condrestart`	Dr.Web ConfigDが実行している場合のみ、再起動するように指示します。
`reload`	コンポーネントが実行している場合は、HUPシグナルをDr.Web ConfigDに送信するように指示します。Dr.Web ConfigDは、このシグナルをDr.Web for UNIX Internet Gatewaysのすべてのコンポーネントに転送します。このパラメータは、すべてのコンポーネントに設定を再度読み込ませるために使用されます。
`status`	現在のDr.Web ConfigDの状態をコンソールに出力するように指示します。

たとえば、**GNU/Linux** OSでDr.Web for UNIX Internet Gatewaysを再起動(または実行されていない場合は起動)するには、次のコマンドを使用します。

```
# /etc/init.d/drweb-configd restart
```

集中管理サーバーに接続する方法

1. 集中管理サーバーのアドレスとその証明書のファイルをアンチウイルスネットワーク管理者から入手します。またワークステーションのIDとパスワードやメイングループと課金プライングループのIDなどの追加パラメータが必要になる場合があります。
2. Dr.Web for UNIX Internet Gatewaysで提供されるDr.Web Ctlコマンドラインツールの`esconnect`コマンドを使用します。

   接続するには、サーバーの証明書ファイルへのパスを指定して、`--Certificate`オプションを使用する必要があります。`--Login`と`--Password`パラメータを使用して、ホストのID(集中管理サーバーで使用される用語を使用する場合は「ワークステーション」ID)と集中管理サーバーの認証用パスワードも入力できます。この場合、サーバーへの接続は、正しいIDとパスワードのペアを指定した場合にのみ確立されます。パラ

メータが指定されない場合、サーバーへの接続は、(サーバーの設定に応じて、自動的またはアンチウイルスネットワークの管理者によって)サーバーで承認されている場合にのみ確立されます。

さらに、 `--Newbie`オプション(新しいユーザーとして接続する)を使用することもできます。このモードがサーバーで許可されている場合、この接続が承認されると、サーバーは自動的に一意の識別子とパスワードのペアを生成します。これは、このエージェントがサーバーに接続する際に使用されます。このモードでは、このホスト用にすでに別のアカウントが存在している場合でも、集中管理サーバーはそのホストの新しいアカウントを生成します。

集中管理サーバーに接続する際に、Dr.Web for UNIX Internet Gatewaysで実行するコマンドの例:

```
# drweb-ctl esconnect <server address> --Certificate <証明書ファイルへのパス>
```

集中管理サーバーへの接続を確立すると、サーバーに設定されている権限とDr.Web ES Agentコンポーネントの設定パラメータ**MobileMode**の値に応じて、Dr.Web for UNIX Internet Gatewaysは集中管理モードまたはモバイルモードで動作します。無条件にモバイルモードを使用できるようにするには、パラメータの値を`On`に設定します。集中管理モードで動作させるには、パラメータの値を`Off`に設定します。

集中管理サーバーに接続されているDr.Web for UNIX Internet Gatewaysにモバイルモードへの切り替えを指示するコマンドの標準的な例は次のとおりです。

```
# drweb-ctl cfset ESAgent.MobileMode On
```

使用する集中管理サーバーがモバイルモードをサポートしていない、または許可していない場合、**MobileMode**パラメータを調整してもDr.Web for UNIX Internet Gatewaysの動作をモバイルモードに切り替えることはできません。

集中管理サーバーから切断する方法

Dr.Web for UNIX Internet Gatewaysを集中管理サーバーから切断してその動作をスタンドアロンモードに切り替えるには、Dr.Web for UNIX Internet GatewaysにあるDr.Web Ctlコマンドラインツールの`esdisconnect`コマンドを使用します。

```
# drweb-ctl esdisconnect
```

Dr.Web for UNIX Internet Gatewaysをスタンドアロンモードで使用するには、有効なライセンスキーファイルが必要です。それ以外の場合は、動作がスタンドアロンモードに切り替えられた後、Dr.Web for UNIX Internet Gatewaysのアンチウイルス機能が*ブロック*されます。

Dr.Web for UNIX Internet Gatewaysを有効化する方法

1. Doctor WebのWebサイト https://products.drweb.com/register/ から登録を実施します。
2. 登録時に指定したメールアドレスに、有効なライセンスキーファイルを含むアーカイブが送信されます(登録後にこのアーカイブをWebサイトから直接ダウンロードすることもできます)。
3. キーファイルのインストール手順を実行します。

Dr.Web for UNIX Internet Gatewaysをアップグレードする方法

コンポーネントのバージョンを更新するか、新しいバージョンにアップグレードしてください。

アップグレード中に、現在のDr.Web for UNIX Internet Gatewaysバージョンを削除するように求められることがあります。

Dr.Web for UNIX Internet Gatewaysコンポーネントを追加または削除する方法

コンポーネントのカスタムインストールとアンインストールの手順に従います。

コンポーネントをインストール/アンインストールする場合、依存関係を解消するために他のDr.Web for UNIX Internet Gatewaysコンポーネントを追加でインストールまたはアンインストールすることがあります。

コンポーネント動作を管理する方法

Dr.Web for UNIX Internet Gatewaysコンポーネントのステータスを表示したり、それらの設定を管理したりするには、次のものを使用できます。

- コマンドラインベースの管理ツールDr.Web Ctl(**drweb-ctl** `appinfo`、**drweb-ctl** `cfshow`および**drweb-ctl** `cfset`コマンドを使用します。使用可能なコマンドの一覧を表示するには、**drweb-ctl** `--help`コマンドを使用します)。
- Dr.Web for UNIX Internet Gatewaysの管理用ウェブインターフェース(初期設定では、ウェブブラウザから`https://127.0.0.1:4443/`にアクセスすると利用できます)。

Dr.Web for UNIX Internet Gatewaysのログを表示する方法

デフォルト設定に従って、すべてのDr.Web for UNIX Internet Gatewaysコンポーネントの一般ログは**syslog**ファイルに表示されます(システムコンポーネント**syslog**によってメッセージを記録するファイルはシステムに依存し、ディレクトリ`/var/log`にあります)。一般ログ設定は、設定ファイルの`[Root]`セクション(**Log**パラメータと**DefaultLogLevel**パラメータ)で定義されます。設定セクションの各コンポーネントには、**Log**パラメータと**LogLevel**パラメータがあります。ログの保存場所と、コンポーネントがログに出力するメッセージのログレベルを設定します。

また、**drweb-ctl** `log`コマンドを使用することもできます。

ログ設定を変更するには、コマンドライン管理ツールDr.Web CtlとDr.Web for UNIX Internet Gateways管理Webインターフェース(インストールされている場合)を使用してください。

- エラーを特定するために、すべてのコンポーネントのログの出力先を別のファイルに設定し、デバッグ情報がログに出力されるようにすることを推奨します。そのためには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.Log <path to log file>
# drweb-ctl cfset Root.DefaultLogLevel DEBUG
```

- デフォルトのロギング方法とログレベルに戻すには、次のコマンドを実行します。

```
# drweb-ctl cfset Root.Log -r
# drweb-ctl cfset Root.DefaultLogLevel -r
```

Dr.Web for UNIX Internet Gatewaysコンポーネント

このセクションでは、Dr.Web for UNIX Internet Gatewaysのコンポーネントについて説明します。各コンポーネントの機能、動作原理、設定ファイルに保存されているパラメータに関する情報を確認できます。

Dr.Web ConfigD

Dr.Web ConfigD設定デーモンは、Dr.Web for UNIX Internet Gatewaysのコアコンポーネントです。すべてのDr.Web for UNIX Internet Gatewaysコンポーネント用に設定情報の中央ストレージを提供し、すべてのコンポーネントの動作を管理し、コンポーネント間の信頼できるデータの交換を整理します。

動作原理

メイン機能

1. 設定に応じてDr.Web for UNIX Internet Gatewaysンポーネントを起動／停止します。動作に障害が発生した場合、コンポーネントを自動的に再起動します。他のコンポーネントのリクエストに応じてコンポーネントを起動します。別のコンポーネントが起動またはシャットダウンしたときに、アクティブなDr.Web for UNIX Internet Gatewaysコンポーネントに通知します。
2. 構成設定へのすべてのコンポーネントの集中アクセスを提供します。設定パラメータを集中管理するためのインターフェースを持つ特別なコンポーネントを提供します。設定の変更についてすべての必須コンポーネントに通知します。
3. 使用されているライセンスキーファイルからの情報をコンポーネントに提供します。特別なコンポーネントから新しいライセンス情報を受け取ります。ライセンスデータまたは構成パラメータの変更について、コンポーネントを実行しているDr.Web for UNIX Internet Gatewaysに通知します。

設定デーモンDr.Web ConfigDは常にroot権限で起動されます。他のDr.Web for UNIX Internet Gatewaysコンポーネントを起動し、事前に開いているソケットを介してそれらと通信します。設定デーモンは、情報ソケット（公に利用可能）と管理ソケット（スーパーユーザー権限を持つコンポーネントのみ利用可能）を介して他のDr.Web for UNIX Internet Gatewaysコンポーネントから接続を受け取ります。設定パラメータとライセンスデータをファイルから読み込む、またはDr.Web ES Agentを介して使用されている集中管理サーバーから配信し、設定パラメータをデフォルトの正しい値に置き換えます。そのため、いずれかのコンポーネントが起動したとき、またはSIGHUPシグナルが送信されたとき、設定デーモンにはDr.Web for UNIX Internet Gatewaysの完全な一貫したパラメータのセットが設定されます。

SIGHUPシグナルを受信すると、設定管理デーモンは設定パラメータとライセンスデータを再度読み込みます。必要に応じて、デーモンはすべてのコンポーネント通知を送信して、設定を再度読み込むように指示します。SIGTERMシグナルを受信すると、デーモンはすべてのコンポーネントをシャットダウンしてから、自身の操作を終了します。デーモンは、シャットダウン後にコンポーネントの一時ファイルもすべて削除します。

通信の原理

1. すべてのコンポーネントは、起動時に設定デーモンDr.Web ConfigDから受信した設定パラメータとライセンスデータのみを使用します。

2. デーモンは、管理されているすべてのコンポーネントから統合ログにメッセージを収集します。コンポーネントからエラーストリーム*stderr*に出力されたすべての情報はデーモンによって収集され、どのコンポーネントがこれを出力したかを示すマークとともにDr.Web for UNIX Internet Gatewaysの統合ログに書き込まれます。
3. シャットダウンすると、管理されているコンポーネントは終了コードを返します。コードが101、102、または103ではない場合、設定デーモンはこのコンポーネントを再起動します。そのため、コンポーネントが異常終了すると再起動され、*stderr*からのエラーメッセージがDr.Web for UNIX Internet Gatewaysのログに登録されます。
   - コンポーネントがコード101で終了した場合、コンポーネントはライセンスパラメータが変更された後にのみ再起動されます。そのため、ライセンスの制限のためにコンポーネントが操作できない場合、コンポーネントはその操作を終了してコード101を*stderr*に出力します。
   - コンポーネントがコード102で終了した場合、そのコンポーネントは設定パラメータが変更された後にのみ再起動されます。そのため、設定が原因でコンポーネントが操作できない場合、コンポーネントはその操作を終了してコード102を*stderr*に出力します。設定デーモンは、パラメータが変更された後にのみコンポーネントの再起動を試みます。
   - リクエストに応じて設定デーモンによって開始されたコンポーネントは、アイドル状態になったときにその動作を終了し、コード103を出力することがあります。たとえば、Dr.Web Scanning Engine、Dr.Web File Checkerなどのコンポーネントです。
   - 設定デーモンからコンポーネントが受信した新しいパラメータ値を「オンザフライ」で適用できない場合、つまり再起動が必要な場合、コンポーネントはコード0で終了します。その場合、Dr.Web ConfigDはコンポーネントを再起動します。
   - コンポーネントが設定デーモンに接続できない、または通信プロトコルエラーが発生した場合、コンポーネントは*stderr*に適切なメッセージを出力してコード1で終了します。
4. シグナル交換:
   - 設定デーモンは、コンポーネントに`SIGHUP`シグナルを送信して設定パラメータを変更するよう指示します。
   - 設定デーモンは、コンポーネントに`SIGTERM`シグナルを送信して30秒以内に操作を終了するよう指示します。
   - `SIGKILL`シグナルは設定デーモンから送信され、`SIGTERM`シグナルの受信後30秒以内にシャットダウンしなかったコンポーネントを強制的に停止させます。

コマンドライン引数

設定デーモンDr.Web ConfigDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-configd [<parameters>]
```

設定デーモンDr.Web ConfigDは、以下のオプションを処理できます。

パラメータ	説明
`--help`	**機能**:コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**: `-h` **引数**: None
`--version`	**機能**:このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。

	短縮形： `-v` **引数**： None
`--config`	**説明**：指定した設定ファイルを今後の操作に使用するように指示します。 **短縮形**： `-c` **引数**： *<path to the file>* - 使用する設定ファイルへのパス。
`--daemonize`	**説明**：コンポーネントをデーモンとして実行するように指示します。つまり、端末にはアクセスできません。 **短縮形**： `-d` **引数**： None
`--pid-file`	**説明**：指定されたPIDファイルを今後の操作に使用するように指示します。 **短縮形**： `-p` **引数**： *<path to the file>* - プロセスID（PID）の保存先ファイルへのパス。

例：

```
$ /opt/drweb.com/bin/drweb-configd -d -c /etc/opt/drweb.com/drweb.ini
```

このコマンドはDr.Web ConfigDを`/etc/opt/drweb.com/drweb.ini`の設定ファイルを使用するデーモンとして実行します。

スタートアップノート

Dr.Web for UNIX Internet Gatewaysの操作を有効にするには、Dr.Web ConfigDがデーモンとして実行されている必要があります。標準起動中、Dr.Web ConfigDはOSの起動時に自動的に起動されます。そのため、Dr.Web ConfigDは、標準のOSディレクトリ（**GNU/Linux**の場合は`/etc/init.d/`、**FreeBSD**の場合は`/usr/local/etc/rc.d/`）にある標準管理スクリプト`drweb-configd`と一緒に保存されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-configd`

設定パラメータ

デーモンのDr.Web ConfigDは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[Root]`セクションにあるコンフィギュレーションパラメータを使用します。

セクションには以下のパラメータが含まれています。

`DefaultLogLevel` *{logging level}*	すべてのDr.Web for UNIX Internet Gatewaysコンポーネントのイベントログのデフォルトのログレベルを定義します。 *このパラメータの値は、設定で独自のログレベルが設定されていない製品内のすべてのコンポーネントに使用されます。* **デフォルト値**：`Notice`

LogLevel *{logging level}*	Dr.Web ConfigDのイベントログのロギングレベル。 **デフォルト値**：Notice
Log *{log type}*	設定デーモンのロギング方式と、このパラメータに別の値が指定されていないコンポーネントのロギング方式。 設定ファイルが読み込まれる前の初回起動時に、設定デーモンは次のパラメータ値を使用します。 • デーモンとして（-dオプションを付けて実行した場合）- SYSLOG:Daemon • その他の場合 - Stderr コンポーネントがバックグラウンドモードで動作している（コマンドラインから-dオプションを使用して起動した）場合は、Stderrの値をこのパラメータに*使用することはできません*。 **デフォルト値**：Syslog:Daemon
PublicSocketPath *{path to file}*	すべてのDr.Web for UNIX Internet Gatewaysコンポーネント間のやり取りに使用されるソケットへのパス。 **デフォルト値**：/var/run/.com.drweb.public
AdminSocketPath *{path to file}*	昇格された（管理者）権限を持つDr.Web for UNIX Internet Gatewaysコンポーネント間のやり取りに使用されるソケットへのパス。 **デフォルト値**：/var/run/.com.drweb.admin
CoreEnginePath *{path to file}*	Dr.Web Virus-Finding Engineスキャンエンジンの動的ライブラリへのパス。 **デフォルト値**：*<var_dir>*/lib/drweb32.dll • **GNU/Linux**の場合：/var/opt/drweb.com/lib/drweb32.dll • **FreeBSD**の場合： /var/drweb.com/lib/drweb32.dll
VirusBaseDir *{path to directory}*	ウイルスデータベースファイルがあるディレクトリへのパス。 **デフォルト値**：*<var_dir>*/bases • **GNU/Linux**の場合：/var/opt/drweb.com/bases • **FreeBSD**の場合： /var/drweb.com/bases
KeyPath *{path to file}*	キーファイルへのパス（正規またはデモライセンス）。 **デフォルト値**：*<etc_dir>*/drweb32.key • **GNU/Linux**の場合：/etc/opt/drweb.com/drweb32.key • **FreeBSD**の場合：/usr/local/etc/drweb.com/drweb32.key
CacheDir *{path to directory}*	キャッシュディレクトリへのパス（更新されたキャッシュとスキャンされたファイルに関する情報のキャッシュを保持するために使用されます）。 **デフォルト値**：*<var_dir>*/cache • **GNU/Linux**の場合：/var/opt/drweb.com/cache • **FreeBSD**の場合： /var/drweb.com/cache
TempDir	一時ファイルがあるディレクトリへのパス。

{path to directory}	**デフォルト値**：*システム環境変数*TMPDIR、TMP、TEMPまたはTEMPDIR *からコピーされたパス（環境変数はこの順序で検索されます）。これらの環境変数がない場合は、*/tmp*。*
RunDir *{path to directory}*	実行中のコンポーネントが有するすべてのPIDファイルと、Dr.Web for UNIX Internet Gatewaysコンポーネント間のやり取りに使用されるソケットを含むディレクトリへのパス。 **デフォルト値**：/var/run
VarLibDir *{path to directory}*	Dr.Web for UNIX Internet Gatewaysコンポーネントによって使用されるライブラリを含むディレクトリへのパス。 **デフォルト値**：*<var_dir>*/lib • **GNU/Linux**の場合：/var/opt/drweb.com/lib • **FreeBSD**の場合：/var/drweb.com/lib
VersionDir *{path to directory}*	Dr.Web for UNIX Internet Gatewaysコンポーネントの現在のバージョンに関する情報が格納されているディレクトリへのパス。 **デフォルト値**：*<var_dir>*/version • **GNU/Linux**の場合：/var/opt/drweb.com/version • **FreeBSD**の場合：/var/drweb.com/version
DwsDir *{path to directory}*	インターネットリソースカテゴリーの自動的に更新されるデータベースのファイルを含むディレクトリへのパス。 **デフォルト値**：*<var_dir>*/dws • **GNU/Linux**の場合：/var/opt/drweb.com/dws • **FreeBSD**の場合：/var/drweb.com/dws
AdminGroup *{group name | GID}*	Dr.Web for UNIX Internet Gateways管理用の管理者権限を持つユーザーのグループ。*root*スーパーユーザーに加えて、これらのユーザーはDr.Web for UNIX Internet Gatewaysコンポーネントの権限をスーパーユーザー権限に昇格させることができます。 **デフォルト値**：*Dr.Web for UNIX Internet Gatewaysのインストール中に決定されます。*
TrustedGroup *{group name | GID}*	信頼するユーザーのグループ。このパラメータはネットワークトラフィックモニターコンポーネント（SpIDer Gate）の動作に使用されます。これらのユーザーのネットワークトラフィックはスキャンされずにSpIDer Gateによってスキップされます。 *ここには存在しないグループを指定することはできません。その場合、SpIDer Gateは起動に失敗します。* *パラメータ値がない場合は、SpIDer Gate設定の***OutputDivert***パラメータに*Auto*値を指定することはできません。* **デフォルト値**：drweb
DebugIpc *{Boolean}*	詳細なIPCメッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。IPCメッセージは、設定デーモンと他のコンポーネントとの間のやり取りを示します。 **デフォルト値**：No

UseCloud *{Boolean}*	Dr.Web Cloudサービスを参照して悪意のあるファイルやURLに関する情報を受け取るかどうかを指定します **デフォルト値**：No
AntispamCorePath *{path to file}*	このパラメータは使用されません。 **デフォルト値**：*<var_dir>*/lib/vaderetro.so • **GNU/Linux**の場合：/var/opt/drweb.com/lib/vaderetro.so • **FreeBSD**の場合：/var/drweb.com/lib/vaderetro.so
VersionNotification *{Boolean}*	現在インストールされているDr.Web for UNIX Internet Gatewaysバージョンのアップデートが利用可能であることをユーザーに通知します。 **デフォルト値**：Yes

Dr.Web Ctl

このセクションの内容：

概要

特別なDr.Web Ctlユーティリティ（**drweb-ctl**）を使用することで、OSのコマンドラインからDr.Web for UNIX Internet Gateways の動作を管理できます。このユーティリティを使用して次の動作を実行できます。

- ブートレコードを含む、ファイルシステムオブジェクトのスキャンを開始する
- リモートネットワークホストでファイルのスキャンを開始する（下記 の注を参照）
- アンチウイルスコンポーネント（ディストリビューションに応じてウイルスデータベース、スキャンエンジンなど）の更新を開始する
- Dr.Web for UNIX Internet Gateways設定のパラメータを確認・変更する
- Dr.Web for UNIX Internet Gatewaysコンポーネントのステータスや検出された脅威に関する統計を確認する
- 集中管理サーバーに接続、または集中管理サーバーとの接続を切断する
- 隔離されたオブジェクトを確認・管理する（Dr.Web File Checkerコンポーネントを介して）
- 集中管理サーバーに接続、または集中管理サーバーとの接続を切断する

Dr.Web for UNIX Internet Gatewaysを管理するためのユーザーコマンド は Dr.Web ConfigD 設定デーモンが動作中の場合のみ適用されます（デフォルトでは、このコンポーネントはシステム起動時に自動的に起動します）。

一部のコントロールコマンドはスーパーユーザー権限を必要とします。

権限を昇格させるには**su**コマンド（カレントユーザーを変更する）または**sudo**コマンド（指定したコマンドを他のユーザーの権限で実行する）を使用します。

drweb-ctlツールはDr.Web for UNIX Internet Gatewaysの動作を管理するコマンドの自動補完をサポートしています（コマンドシェル内で該当するオプションが有効になっている場合）。コマンドシェルが自動補完を許可していない場合、このオプションの設定を行うことができます。方法については、お使いのOSディストリビューションのマニュアルを参照してください。

シャットダウンする際、ツールはPOSIX準拠システムの規則に従って終了コードを返します。操作が正常に完了した場合は0（ゼロ）、それ以外の場合は0以外（ゼロ以外）です。

ツールが0以外（non-null）の終了コードを返すのは、内部エラーの場合のみであるという点に注意してください（例：ツールがコンポーネントに接続できなかった、リクエストされた操作を実行できなかった）。ツールが脅威を検出（そして駆除）した場合は、リクエストされた操作（例：スキャン）が正常に実行されたため、0（null）終了コードを返します。検出された脅威と適用されたアクションのリストを明らかにする必要がある場合、コンソールに表示されたメッセージを分析してください。

すべてのエラーのコードについては 付録F. 既知のエラー セクションのリストをご確認ください。

リモートホストスキャン

Dr.Web for UNIX Internet Gatewaysを使用して、リモートネットワークホストにあるファイルの脅威に対するスキャンを実行できます。このようなホストには、フルコンピューティングマシン（ワークステーションやサーバーなど）だけでなく、ルーター、セットトップボックス、いわゆる「モノのインターネット（IoT）」と呼ばれるその他の「スマート」デバイスも含まれます。リモートスキャンを実行するには、リモートホストが*SSH*（*セキュアシェル*）または*Telnet*を介したリモート端末アクセスを提供する必要があります。デバイスにアクセスするには、リモートホストのIPアドレスとドメイン名、*SSH*または*Telnet*を介してリモートでシステムにアクセスするユーザーの認証情報を知っている必要があります。このユーザーは、スキャン済みファイルへのアクセス権限（少なくとも読み取り権限）を持っている必要があります。

この機能は、リモートホスト上の悪質なファイルや疑わしいファイルの検出にのみ使用できます。リモートスキャンの手段を用いた脅威の排除（すなわち、悪意のあるオブジェクトの隔離への移動、削除および修復）はできません。リモートホスト上で検出された脅威を排除するには、このホストが直接提供する管理ツールを使用する必要があります。たとえば、ルーターおよび他の「スマート」デバイスの場合、ファームウェア更新のためのメカニズムを使用できます。コンピューティングマシンの場合、それらへの接続（たとえば、リモートターミナルモードを使用）、ファイルシステム内のそれぞれの操作（ファイルの削除または移動など）、またはそれらにインストールされたアンチウイルスソフトウェアの実行により行うことができます。

リモートスキャンはコマンドラインツール **drweb-ctl** からのみ実行できます（コマンド `remotescan` を使用します）。

コマンドラインフォーマット

1.製品を管理するためのコマンドラインユーティリティのコマンドフォーマット

Dr.Web for UNIX Internet Gateways の動作を管理するコマンドラインツールのフォーマットは以下のとおりです。

```
$ drweb-ctl [<general options> | <command> [<argument>] [<command options>]]
```

- *<general options>* - コマンドが指定されていない場合に起動時に適用できる、またはあらゆるコマンドにおいて適用できるオプションです。起動時に必須ではありません。
- *<command>* - Dr.Web for UNIX Internet Gatewaysによって実行されるコマンドです（スキャンの開始、隔離されたオブジェクトのリストを出力、その他のコマンドなど）。
- *<argument>* - コマンド引数です。指定されたコマンドに依存します。コマンドによってはない場合もあります。
- *<command options>* - 指定されたコマンドの動作を管理するためのオプションです。一部のコマンドでは省略できます。

2. 全般的なオプション

以下の全般的なオプションを使用できます。

オプション	説明
`-h, --help`	全般的なヘルプ情報を表示して終了します。いずれかのコマンドに関するヘルプ情報を表示させるには、以下の呼び出しを使用します。 `$ drweb-ctl <command> -h`
`-v, --version`	モジュールバージョンに関する情報を表示して終了します。
`-d, --debug`	指定されたコマンドの実行時にデバッグ情報を表示するよう指示します。コマンドが指定されていない場合は実行できません。以下の呼び出しを使用します。 `$ drweb-ctl <command> -d`

3. コマンド

Dr.Web for UNIX Internet Gatewaysを管理するコマンドは以下のグループに分けることができます。

- アンチウイルススキャンのコマンド。
- 更新および集中管理モードでの動作を管理するコマンド。
- 設定を管理するコマンド。

- 検出された脅威および隔離を管理するコマンド。
- 情報に関するコマンド。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します **man** `1 drweb-ctl`。

3.1. アンチウイルススキャンのコマンド

アンチウイルススキャンを管理するコマンドには以下のものがあります。

コマンド	説明
`scan` *<path>*	**目的**：Dr.Web File Checkerコンポーネントを介して、指定されたファイルまたはディレクトリのチェックを開始します。 **引数**： *<path>*—スキャンの対象として選択されたファイルやディレクトリへのパスです。 *`--stdin` または `--stdin0` オプションを使用する場合、この引数は省略できます。特定の条件を満たす複数のファイルを指定するには、***find*** ユーティリティ（使用例 参照）および`--stdin` または `--stdin0` オプションを使用します。* **オプション**： `-a [--Autonomous]` - Dr.Web Scanning EngineとDr.Web File Checkerの自律コピーを実行して指定したチェックを実行し、チェックが完了した後に終了します。自律スキャン中に検出された脅威は、`threats`コマンド（以下参照）で表示される検出された脅威の共通リストに追加されません。またそれぞれの管理下で製品が実行されている場合、そうした脅威に関する情報は集中管理サーバーに配信されません。 `--stdin`—スキャンのためのパスのリストを標準的な入力文字列（*stdin*）から取得します。リスト内のパスは改行文字（`\n`）で区切られている必要があります。 `--stdin0`—スキャンのためのパスのリストを標準的な入力文字列（*stdin*）から取得します。リスト内のパスはヌル文字（`\0`）で区切られている必要があります。 > `--stdin` および `--stdin0` を使用する際、リスト内のパスには検索のためのパターンまたは正規表現が含まれていないようにする必要があります。`--stdin` および `--stdin0` オプションの推奨される使用方法は、**scan** コマンド内でパスのリスト（`find` などの外部ユーティリティによって生成された）を処理するというものです（使用例 参照）。 `--Exclude` *<path>* - チェックから除外するパス（マスク記号 `*`と`?`は使用できます）。 *任意オプション、複数回設定できます。* `--Report` *<type>* - スキャンレポートのタイプを指定します。 **使用可能な値**： • `BRIEF` - 短いレポート

コマンド	説明
	• DEBUG - 詳細なレポート。 • JSON - JSON形式のシリアル化されたレポート。 **デフォルト値**: *BRIEF* --ScanTimeout <*number*> - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**: *0* --PackerMaxLevel <*number*> - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* --ArchiveMaxLevel <*number*> - アーカイブ(zip、rarなど)をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* --MailMaxLevel <*number*> - メールメッセージ(pst、tbbなど)をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* --ContainerMaxLevel <*number*> - その他のコンテナ(HTMLなど)をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* --MaxCompressionRatio <*ratio*> - スキャンされるオブジェクトの最大圧縮率を指定します。 率は *2* 以上にする必要があります。 **デフォルト値**: *3000* --HeuristicAnalysis <*On*|*Off*> - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値**: *On* --OnKnownVirus <*action*>—シグネチャ解析を用いて検出された脅威に適用されるアクションです。 **使用可能な値**: *REPORT*、*CURE*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* --OnIncurable <*action*> - 検出された脅威の修復に失敗した場合に適用される、または修復不可能な脅威に対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT*

コマンド	説明
	`--OnSuspicious` *<action>* - ヒューリスティック解析によって検出された疑わしいオブジェクトに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**：*REPORT* `--OnAdware` *<action>* - 検出されたアドウェアプログラムに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**：*REPORT* `--OnDialers` *<action>* - ダイアラーに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**：*REPORT* `--OnJokes` *<action>* - ジョークプログラムに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE*。 **デフォルト値**：*REPORT* `--OnRiskware` *<action>* - 潜在的に危険なプログラム（リスクウェア）に対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**：*REPORT* `--OnHacktools` *<action>* - ハッキングツールに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**：*REPORT* コンテナ（アーカイブ、メール添付ファイルなど）内のファイルで脅威が検出された場合は、削除（*DELETE*）アクションの代わりにコンテナの隔離への移動（*QUARANTINE*）が実行されます。
`bootscan` *<disk drive>* | `ALL`	**目的**：Dr.Web File Checkerコンポーネントを介して、指定されたディスクのブートレコードのチェックを開始します。MBRとVBRの両方のレコードがスキャンされます。 **引数**： *<disk drive>* - ブートレコードをスキャンするディスクデバイスのブロックファイルへのパス。スペースで区切って複数のディスクデバイスを指定できます。引数は必須です。デバイスファイルの代わりに`ALL`を指定した場合は、使用可能な全てのディスクデバイスにある全てのブートレコードが確認されます。 **オプション**： `-a [--Autonomous]` - Dr.Web Scanning EngineとDr.Web File Checkerの自律コピーを実行して指定したチェックを実行し、チェックが完了した後に終了します。自律スキャン中に検出された脅威は、`threats`コマンド（以下参照）で表示される検出された脅威の共通リストに追加されません。またそれぞれの管理下で製品が実行されている場合、そうした脅威に関する情報は集中管理サーバーに配信されません。 `--Report` *<type>* - レポートのタイプを指定します。

コマンド	説明
	使用可能な値: • BRIEF - 短いレポート • DEBUG - 詳細なレポート。 • JSON - JSON形式のシリアル化されたレポート。 **デフォルト値**: *BRIEF* --ScanTimeout *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**: *0* --HeuristicAnalysis *<On|Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値**: *On* --Cure *<Yes|No>* - 脅威が検出された際に修復を試みる動作を有効または無効にします。 値に *No* が指定された場合、検出された脅威に関する通知のみが表示されます。 **デフォルト値**: *No* --ShellTrace - ブートレコードをスキャンする際の、追加のデバッグ情報の表示を有効にします。
procscan	**目的**: Dr.Web File Checkerを使用して、現在実行中のプロセスのコードを含む実行可能ファイルのチェックを開始します。悪意のある実行可能ファイルが検出されると、そのファイルは無効化され、このファイルによって実行される全てのプロセスは強制的に終了されます。 **引数**: None **オプション**: -a [--Autonomous] - Dr.Web Scanning EngineとDr.Web File Checkerの自律コピーを実行して指定したチェックを実行し、チェックが完了した後に終了します。自律スキャン中に検出された脅威は、threatsコマンド(以下参照)で表示される検出された脅威の共通リストに追加されません。またそれぞれの管理下で製品が実行されている場合、そうした脅威に関する情報は集中管理サーバーに配信されません。 --Report *<type>* - レポートのタイプを指定します。 **使用可能な値**: • BRIEF - 短いレポート • DEBUG - 詳細なレポート。 • JSON - JSON形式のシリアル化されたレポート。 **デフォルト値**: *BRIEF* --ScanTimeout *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**: *0*

コマンド	説明
	`--HeuristicAnalysis` *<On|Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値：** *On* `--PackerMaxLevel` *<number>* - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値：** *8* `--OnKnownVirus` *<action>*—シグネチャ解析を用いて検出された脅威に適用されるアクションです。 **使用可能な値：** *REPORT*、*CURE*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnIncurable` *<action>* - 検出された脅威の修復に失敗した場合に適用される、または修復不可能な脅威に対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnSuspicious` *<action>* - ヒューリスティック解析によって検出された疑わしいオブジェクトに対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnAdware` *<action>* - 検出されたアドウェアプログラムに対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnDialers` *<action>* - ダイアラーに対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnJokes` *<action>* - ジョークプログラムに対して適用されるアクションです。 **使用可能な値**：*REPORT*、*QUARANTINE*、*DELETE*。 **デフォルト値：** *REPORT* `--OnRiskware` *<action>* - 潜在的に危険なプログラム（リスクウェア）に対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* `--OnHacktools` *<action>* - ハッキングツールに対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT*

コマンド	説明
	実行可能ファイルで脅威が検出された場合、Dr.Web for UNIX Internet Gatewaysはそのファイルから起動した全てのプロセスを終了させます。
`netscan [<path>]`	**目的**：ネットワークデータスキャン用にDr.Web Network Checkerエージェントを介して、指定されたファイルまたはディレクトリの分散スキャンを開始します。Dr.Web for UNIXを実行している他のホストへの接続が設定されていない場合、スキャンはローカルで利用可能なScanning Engine（`scan`コマンドと同様）を介してのみ行われます。 **引数**： *<path>* - スキャンの対象として選択されたファイルやディレクトリへのパスです。 *この引数を省略すると、入力スレッドstdinを介して受信したデータがスキャンされます。* **オプション**： `--Report` *<type>* - スキャンレポートのタイプを指定します。 **使用可能な値**： • `BRIEF` - 短いレポート • `DEBUG` - 詳細なレポート。 • `JSON` - JSON形式のシリアル化されたレポート。 **デフォルト値**：*BRIEF* `--ScanTimeout` *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**：*0* `--HeuristicAnalysis` *<On/Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値**：*On* `--PackerMaxLevel` *<number>* - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--ArchiveMaxLevel` *<number>* - アーカイブ（zip、rarなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--MailMaxLevel` *<number>* - メールメッセージ（pst、tbbなど）をスキャンする際のネスティングレベルの上限を指定します。

コマンド	説明
	値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--ContainerMaxLevel` *<number>* - その他のコンテナ（HTMLなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--MaxCompressionRatio` *<ratio>* - スキャンされるオブジェクトの最大圧縮率を指定します。 率は *2* 以上にする必要があります。 **デフォルト値**: *3000* `--Cure` *<Yes/No>* - 脅威が検出された際に修復を試みる動作を有効または無効にします。 値に *No* が指定された場合、検出された脅威に関する通知のみが表示されます。 **デフォルト値**: *No*
`flowscan` *<path>*	**目的**: Dr.Web File Checkerを使用して"flow"メソッドを用いて指定されたファイルまたはディレクトリのスキャンを開始します ⚠ ファイルとディレクトリのオンデマンドスキャンの場合は、`scan`コマンドを使用することを推奨します。 **引数**: *<path>* - スキャンの対象として選択されたファイルやディレクトリへのパスです。 **オプション**: `--ScanTimeout` *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**: *0* `--HeuristicAnalysis` *<On/Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値**: *On* `--PackerMaxLevel` *<number>* - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--ArchiveMaxLevel` *<number>* - アーカイブ（zip、rarなど）をスキャンする際のネスティングレベルの上限を指定します。

コマンド	説明
	値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--MailMaxLevel` *<number>* - メールメッセージ（pst、tbbなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--ContainerMaxLevel` *<number>* - その他のコンテナ（HTMLなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**: *8* `--MaxCompressionRatio` *<ratio>* - スキャンされるオブジェクトの最大圧縮率を指定します。 率は *2* 以上にする必要があります。 **デフォルト値**: *3000* `--OnKnownVirus` *<action>*—シグネチャ解析を用いて検出された脅威に適用されるアクションです。 **使用可能な値**: *REPORT*、*CURE*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* `--OnIncurable` *<action>* - 検出された脅威の修復に失敗した場合に適用される、または修復不可能な脅威に対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* `--OnSuspicious` *<action>* - ヒューリスティック解析によって検出された疑わしいオブジェクトに対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* `--OnAdware` *<action>* - 検出されたアドウェアプログラムに対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* `--OnDialers` *<action>* - ダイアラーに対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値**: *REPORT* `--OnJokes` *<action>* - ジョークプログラムに対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE*。 **デフォルト値**: *REPORT* `--OnRiskware` *<action>* - 潜在的に危険なプログラム（リスクウェア）に対して適用されるアクションです。 **使用可能な値**: *REPORT*、*QUARANTINE*、*DELETE*

コマンド	説明
	デフォルト値： *REPORT* `--OnHacktools` *<action>* - ハッキングツールに対して適用されるアクションです。 **使用可能な値：** *REPORT*、*QUARANTINE*、*DELETE* **デフォルト値：** *REPORT* コンテナ（アーカイブ、メール添付ファイルなど）内のファイルで脅威が検出された場合は、削除アクション（Delete）の代わりにコンテナの隔離への移動（Quarantine）が実行されます。
`rawscan` *<path>*	**目的：** Dr.Web File Checkerを使用せず、Dr.Web Scanning Engineに指定されたファイルまたはディレクトリに対して"raw"スキャンを直接実行します。 "raw"スキャンで検出された脅威は、`threats`コマンドで表示される脅威のリストには含まれないということに注意してください（以下参照）。 このコマンドは、Dr.Web Scanning Engineの機能をデバッグするためにのみ使用することを推奨します。ファイルで検出された脅威のうち少なくとも*1つ*の脅威が駆除されている場合、コマンドは「修復済」ステータスを出力します（*全*ての脅威が駆除されているわけではありません）。したがって、徹底したファイルスキャンが必要な場合は、このコマンドを使用することは*推奨しません*。後者の場合は、`scan`コマンドを使用することを推奨します。 **引数：** `<path>` - スキャンの対象として選択されたファイルやディレクトリへのパスです。 **オプション：** `--ScanEngine` *<path>* - Dr.Web Scanning EngineのUNIXソケットへのパスです。指定しない場合、Scanning Engineの自律インスタンスが起動されます（スキャンが完了するとシャットダウンします）。 `--Report` *<type>* - レポートのタイプを指定します。 **使用可能な値：** • `BRIEF` - 短いレポート • `DEBUG` - 詳細なレポート。 • `JSON` - JSON形式のシリアル化されたレポート。 **デフォルト値：** *BRIEF* `--ScanTimeout` *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に *0* が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値：** *0*

コマンド	説明
	`--PackerMaxLevel` *<number>* - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値** *: 8* `--ArchiveMaxLevel` *<number>* - アーカイブ（zip、rarなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値** *: 8* `--MailMaxLevel` *<number>* - メールメッセージ（pst、tbbなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値** *: 8* `--ContainerMaxLevel` *<number>* - その他のコンテナ（HTMLなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値** : 8 `--MaxCompressionRatio` *<ratio>* - スキャンされるオブジェクトの最大圧縮率を指定します。 率は *2* 以上にする必要があります。 **デフォルト値** *: 3000* `--HeuristicAnalysis` *<On|Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値** : *On* `--Cure` *<Yes|No>* - 脅威が検出された際に修復を試みる動作を有効または無効にします。 値に *No* が指定された場合、検出された脅威に関する通知のみが表示されます。 **デフォルト値** *: No* `--ListCleanItem` - スキャンされたコンテナで見つかったクリーン（感染していない）なファイルのリストを出力できるようにします。 `--ShellTrace`—ファイルをスキャンする際の、追加のデバッグ情報の表示を有効にします。
`remotescan` *<host> <path>*	**目的**：指定されたリモートホストに接続し、SSHを使用して、指定されたファイルまたはディレクトリのスキャンを開始します。

コマンド	説明
	 リモートスキャンで検出された脅威は駆除されず、`threats` コマンドで表示される脅威のリストには含まれないということに注意してください（下記 参照）。 --- この機能は、リモートホストの悪意のあるファイルや疑わしいファイルの検出にのみ使用できます。リモートホストで検出された脅威を排除するには、このホストによって直接提供される管理ツールを使用する必要があります。たとえば、ルーターやその他の「スマート」デバイスの場合は、ファームウェア更新のメカニズムを使用できます。コンピューティングマシンの場合は、コンピューティングマシンへの接続（任意でリモート端末モードを使用）とファイルシステムのそれぞれの操作（ファイルの削除または移動など）、またはコンピューティングマシンにインストールされているアンチウイルスソフトウェアの実行を介して実行できます。 **引数：** *<host>* - リモートホストのIPアドレスまたはドメイン名。 *<path>* - スキャンの対象として選択されたファイルやディレクトリへのパスです。 **オプション：** `-m [--Method]` *<SSH|Telnet>* - リモートホスト接続方法（プロトコル）。 *方法が指定されていない場合は、SSHが使用されます。* `-l [--Login]` *<name>* - 選択されたプロトコル経由でリモートホストでの認証に使用されるログインID（ユーザー名）。 *ユーザー名が指定されていない場合、コマンドを起動したユーザー名を用いてリモートホストに接続しようとします。* `-i [--Identity]` *<path to file>* - 選択されたプロトコル経由で指定されたユーザーの認証に使用されるプライベートキーが含まれるファイルへのパス。 `-p [--Port]` *<number>* - 選択されたプロトコル経由で接続するリモートホストのポート番号。 **デフォルト値**：選択したプロトコル用の*デフォルトポート（SSHは22、Telnetは23）* `--Password` *<password>* - 選択されたプロトコルを介してユーザー認証に使用されるパスワード。 *パスワードはプレーンテキストとして転送されることに注意してください。* `--Exclude` *<path>* - チェックから除外するパス（マスク記号 `*`と`?`は使用できます）。 *任意オプション、複数回設定できます。* `--Report` *<type>* - スキャンレポートのタイプを指定します。 **使用可能な値：** • `BRIEF` - 短いレポート • `DEBUG` - 詳細なレポート。 • `JSON` - JSON形式のシリアル化されたレポート。 **デフォルト値**：*BRIEF*

コマンド	説明
	`--ScanTimeout` *<number>* - 1つのファイルをスキャンする際のタイムアウトをミリ秒で指定します。 値に 0 が指定された場合、スキャンにかかる時間は制限されません。 **デフォルト値**：0 `--PackerMaxLevel` *<number>* - パックされたオブジェクトをスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--ArchiveMaxLevel` *<number>* - アーカイブ（zip、rarなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--MailMaxLevel` *<number>* - メールメッセージ（pst、tbbなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--ContainerMaxLevel` *<number>* - その他のコンテナ（HTMLなど）をスキャンする際のネスティングレベルの上限を指定します。 値に *0* が指定された場合、ネストされたオブジェクトはスキャン中にスキップされます。 **デフォルト値**：*8* `--MaxCompressionRatio` *<ratio>* - スキャンされるオブジェクトの最大圧縮率を指定します。 率は *2* 以上にする必要があります。 **デフォルト値**：*3000* `--HeuristicAnalysis` *<On|Off>* - スキャン中のヒューリスティック解析を有効または無効にします。 **デフォルト値**： `On`

3.2.更新および集中管理モードでの動作を管理するコマンド

更新および集中管理モード。

コマンド	説明
`update`	**目的**：Doctor Webの更新サーバーまたはDr.Web MeshDを経由したローカルクラウドから、アンチウイルスコンポーネント（ディストリビューションによってはウイルスデータベース、アンチウイルスエンジンなど）の更新プロセスを開始するように指示し、既に実行されている更新プロセスを終了するか、最後の更新を以前に更新したファイルのバージョンにロールバックします。

コマンド	説明
	Dr.Web for UNIX Internet Gatewaysが集中管理サーバーに接続されている場合、このコマンドは無効です。 **引数**: None **オプション**: `-l [--local-cloud]` - Dr.Web MeshDに接続されているローカルクラウドを使用して更新をダウンロードします。オプションが指定されていない場合、更新はDr.Web更新サーバーからダウンロードされます(デフォルト)。 `--Rollback` - 最終更新をロールバックし、最終更新時に更新された前のバージョンのファイルを復元します。 `--Stop` - 実行中の更新プロセスを終了します。
`esconnect` *<server>*`[:`*<port>*`]`	**目的**: Dr.Web for UNIX Internet Gatewaysを指定された集中管理サーバー(Dr.Web Enterprise Serverなど)に接続します。詳細は、動作モードを参照してください。 **引数**: • *<server>* - 集中管理サーバーが動作しているホストのIPアドレスまたはホスト名。この項目は必須です。 • *<port>* - 中管理サーバーによって使用されるポート番号。引数はオプションであり、集中管理サーバーが標準以外のポートを使用する場合にのみ指定する必要があります。 **オプション**: `--Certificate` *<path>*—接続先の集中管理サーバーの証明書へのファイルパス。 `--Login` *<ID>* - 集中管理サーバーへの接続に使用されるログインID(ワークステーションID)。 `--Password` *<password>* - 集中管理サーバーへの接続用パスワード。 `--Group` *<ID>* - ワークステーションが接続時に追加されるグループのID。 `--Rate` *<ID>*—ワークステーションが集中管理サーバーグループの1つに含まれている場合に、そのワークステーションに適用されるタリフグループのID(`--Group`オプションと一緒にのみ指定できます)。 `--Compress` *<On|Off>*—データの圧縮を有効(*On*)または無効(*Off*)にします。指定しない場合、圧縮の使用はサーバーによって決定されます。 `--Encrypt` *<On|Off>*—データの暗号化を有効(*On*)または無効(*Off*)にします。指定しない場合、暗号化の使用はサーバーによって決定されます。 `--Newbie` - "新規端末(newbie)"として接続します(サーバーで新しいアカウントを取得します)。 このコマンドでは、**drweb-ctl**を*ルート*権限で起動する必要があります。必要に応じて、**su**または**sudo**コマンドを使用してください。

コマンド	説明
`esdisconnect`	**目的**：Dr.Web for UNIX Internet Gatewaysを集中管理サーバーから切断し、その動作をスタンドアローン モードに切り替えます。 *Dr.Web for UNIX Internet Gatewaysがスタンドアローンモードで既に動作している場合、このコマンドは無効になります。* **引数**： None **オプション**： None このコマンドでは、**drweb-ctl**をルート権限で起動する必要があります。必要に応じて、**su**または**sudo**コマンドを使用してください。

3.3. 設定を管理するコマンド

設定を管理するコマンドには以下のものがあります。

コマンド	説明
`cfset` *<section>*. *<parameter>* *<value>*	**目的**：現在の設定で、指定されたパラメータのアクティブな値を変更します。 *"="(等号)はできないことに注意してください。* **引数**： • *<section>* - パラメータのある設定ファイルのセクション名です。この引数は必須です。 • *<parameter>* - パラメータの名前です。この引数は必須です。 • *<value>* - パラメータに割り当てられる新しい値です。この引数は必須です *パラメータ値の設定には <section>. <parameter> <value> のフォーマットを常に使用します。* *複数のパラメータ値を指定する場合は、追加するパラメータ値の数だけコマンド`cfset`の呼び出しを繰り返す必要があります。さらに、パラメータ値のリストに新しい値を追加するには、オプション`-a`を使用する必要があります(以下参照)。文字列value1、value2は単一のパラメータ値と見なされるため、コマンドオプション<parameter> value1、value2は使用できません。* *設定ファイルの説明については、付録D. Dr.Web for UNIX Internet Gateways設定ファイルセクション、または***man** 5

コマンド	説明
	drweb.ini で表示されるドキュメントページを参照してください。 **オプション:** `-a [--Add]` - 現在のパラメータ値を置き換えず、指定された値をリストに追加します(リストとして指定された複数の値を持つことのできるパラメータに対してのみ使用可能)。このオプションは、タグを付けたパラメータの新しいグループを追加する場合にも使用してください。 `-e [--Erase]` - 現在のパラメータ値を置き換えず、指定された値をリストから削除します(リストとして指定された複数の値を持つことのできるパラメータに対してのみ使用可能)。このオプションを使用して、タグ付きのパラメータのグループ全体を削除することもできます。 `-r [--Reset]`—パラメータ値をデフォルトにリセットします。その際、コマンド内で *<value>* は必要なく、指定された場合は無視されます。 オプションは必須ではありません。指定されなかった場合は、現在のパラメータ値(パラメータに複数の値がある場合は値の全リスト)が指定された値に置き換えられます。 Dr.Web ClamDモニターでは、個別のパラメータ設定がされているセクションに-rオプションを使用すると、そのパラメータ値は、このコンポーネントの一般設定セクションにある同じ名前の「親」パラメータの値に変更されます。 Dr.Web ClamD用に新しい接続ポイント *<point>* を追加するを追加する必要がある場合は、以下のコマンドを使用します。 `cfset ClamD.Endpoint.<point> -a` `cfset ClamD.Endpoint.point1 -a` 例: このコマンドでは、**drweb-ctl**をルート権限で起動する必要があります。必要に応じて、**su**または**sudo**コマンドを使用してください。
`cfshow` `[ <section>]` `[ . <parameter>]`	**目的:** 現在の設定のパラメータ値を表示します。パラメータは *<section>.<parameter> = <value>* のように画面に出力されます。インストールされていないコンポーネントのセクションとパラメータは表示されません。 **引数:** • *<section>* - 表示するパラメータのある設定セクションの名前です。この引数は任意です。指定されなかった場合、すべての設定セクションのパラメータが表示されます。 • *<parameter>* - 表示するパラメータの名前です。指定されなかった場合、セクションのすべてのパラメータが表示されます。それ以外の場合は、このパラメータのみが表示されます。セクション名なしにパラメータが指定された場合、すべての設定ファイルセクションにある、その名前を持つすべてのパラメータが表示されます。

コマンド	説明
	オプション: `--Uncut`—すべての設定パラメータを出力します（現在インストールされているコンポーネントのセットによって使用されているもの以外も含む）。このオプションが指定されていない場合、インストールされたコンポーネントの設定に使用されているパラメータのみが出力されます。 `--Changed` - デフォルト値とは異なる値を持つパラメータのみを出力します。 `--Ini`—パラメータ値をINIファイルフォーマットで表示します。まず鍵括弧内でセクション名が指定され、次に *<parameter>* = *<value>* ペアでセクションパラメータが表示されます（1行につき1ペア）。 `--Value` - 指定されたパラメータの値のみを出力します（この場合、<parameter>引数は必須です）。
`reload`	**目的**：`SIGHUP`シグナルをDr.Web ConfigD設定デーモンに送信します。 このシグナルを受信すると、Dr.Web ConfigD設定デーモンは設定を再度読み込み、それに必要な変更をDr.Web for UNIX Internet Gatewaysコンポーネントに送信します。その後、設定デーモンはプログラムログを再度開き、ウイルスデータベースを使用するコンポーネント（アンチウイルスエンジンを含む）を再起動し、異常終了したコンポーネントの再起動を試みます。 **引数**： None **オプション**： None

3.4. 検出された脅威および隔離を管理するコマンド

検出された脅威および隔離を管理するコマンドには以下のものがあります。

コマンド	説明
`threats` [*<action>* *<object>*]	**目的**：指定されたアクションを、検出された脅威に適用します。アクションの種類はコマンドのオプションによって指定します。 アクションが指定されていない場合、検出されたが駆除されていない脅威に関する情報を表示します。脅威に関する情報は、オプションの`--Format`引数で指定されたフォーマットに従って表示されます。`--Format`引数が指定されていない場合は、各脅威に関する次の情報が表示されます。 • 脅威に対して割り当てられた識別子（順序数） • 感染したファイルへのフルパス • 脅威に関する情報（脅威の名前、Doctor Webの使用する分類による脅威のタイプ） • ファイルに関する情報（サイズ、ファイル所有者のユーザー名、最後に変更された時間） • 脅威に対して適用された操作の履歴（検出、アクションの適用など） **引数**： None

コマンド	説明
	オプション: `--Format "`*<format string>*`"` - 脅威に関する情報を指定されたフォーマットで表示します。フォーマット文字列の説明は以下のとおりです。 *このオプションを下のオプションと一緒に適用すると無視されます。* `-f [--Follow]` - 新しい脅威に関する新しいメッセージを待ち、それらを受け取り次第、表示します(CTRL+Cで待機を中断します)。 *このオプションがアクションオプションと一緒に指定されている場合は無視されます。* `--Directory` *<list of directories>* - *<list of directories>*で指定したディレクトリ内のファイルで検出された脅威のみを表示します。 *このオプションを下のオプションと一緒に適用すると無視されます。* `--Cure` *<threat list>* - リストアップされた脅威を修復しようと試みます(脅威の識別子をコンマで区切ってリストアップ)。 `--Quarantine` *<threat list>* - リストアップされた脅威を隔離に移します(脅威のIDをコンマで区切ってリストアップ)。 `--Delete` *<threat list>* - リストアップされた脅威を削除します(脅威の識別子をコンマで区切ってリストアップ)。 `--Ignore` *<threat list>* - リストアップされた脅威を無視します(脅威の識別子をコンマで区切ってリストアップ)。 検出されたすべての脅威に対してコマンドを適用する必要がある場合は、*<threat list>*の代わりに `All` を指定します。例: `$ drweb-ctl threats --Quarantine All` この例では、検出された悪意のあるオブジェクトすべてを隔離に移します。
`quarantine [`*<action>* *<object>*`]`	**目的:** 隔離内の指定されたオブジェクトに対してアクションを適用します。 アクションが指定されなかった場合、隔離されたオブジェクトに関する情報とそのIDが、隔離に移された元のファイルに関する簡単な情報と一緒に表示されます。隔離されたオブジェクトに関する情報は、オプションの`--Format`引数で指定されたフォーマットに従って表示されます。`--Format`引数が指定されていない場合は、隔離された各オブジェクト次の情報が表示されます。 • 隔離されたオブジェクトに対して割り当てられた識別子(順序数) • 隔離に移される前の、元のファイルへのパス • ファイルが隔離に移された日付 • ファイルに関する情報(サイズ、ファイル所有者のユーザー名、最後に変更された時間) • 脅威に関する情報(脅威の名前、Doctor Webの使用する分類による脅威のタイプ) **引数:** None **オプション:** `-a [--Autonomous]` - 指定された隔離コマンドを実行するファイルをチェックするためにDr.Web File Checkerコンポーネントの別のインスタンスを起動し、コマンド完了後にシャットダウンします。

コマンド	説明
	このオプションは下のオプションと一緒に適用できます。 `--Format "`*<format string>*`"` - 隔離されたオブジェクトに関する情報を指定されたフォーマットで表示します。フォーマット文字列の説明は以下のとおりです。 *このオプションを下のオプションと一緒に適用すると無視されます。* `-f [--Follow]`—新しい脅威に関する新しいメッセージを待ち、それらを受け取り次第、表示します(CTRL+C で待機を中断します)。 *このオプションがアクションオプションと一緒に指定されている場合は無視されます。* `--Discovery [`*<list of directories>*`,]` - 指定されたディレクトリのリストから隔離ディレクトリを検索し、脅威が検出された場合は統合された隔離ディレクトリに追加します。*<list of directories>*が指定されていない場合は、ファイルシステムの共通の場所(ボリュームのマウントポイントとユーザーのホームディレクトリ)で隔離ディレクトリを検索します。 *このオプションは `-a(--Autonomous)`オプション(上を参照)だけでなく、下に一覧で示されている任意のオプションおよびアクションとともに指定できます。さらに、自律コピーとして `quarantine` コマンドを起動すると(`-a(--Autonomous)`オプションを指定して、`--Discovery` オプションは指定しない場合)、次の呼び出しと同じになります。* `quarantine --Autonomous --Discovery` `--Delete` *<object>* - 指定されたオブジェクトを隔離から削除します。 *オブジェクトは隔離から永久に削除されることに注意してください。この操作は元に戻せません。* `--Cure` *<object>* - 隔離内の指定されたオブジェクトの修復を試みます。 *オブジェクトが修復された場合であっても、それは隔離内に残ります。修復されたオブジェクトを隔離から復元するには `--Restore` コマンドを使用します。* `--Restore` *<object>* - 指定されたオブジェクトを隔離から元の場所に復元します。 *このコマンドでは、**drweb-ctl**をスーパーユーザー権限で起動する必要がある場合があります。感染していても隔離からファイルを復元できます。* `--TargetPath` *<path>* -オブジェクトを隔離から指定された場所に復元します。指定された名前を持つファイルとして復元するか(*<path>*がファイルへのパスであった場合)、またはただ単に指定されたディレクトリに復元します(*<path>*がディレクトリへのパスであった場合)。`--Restore`コマンドとの組み合わせでのみ使用できます。パスは絶対パスでも相対パスでも構いません(現在のディレクトリを参照)。 *<object>* は隔離内のオブジェクトの識別子を指定します。隔離されたすべてのオブジェクトに対してコマンドを適用する場合は、*<object>* の代わりに `All` を指定してください。例: `$ drweb-ctl quarantine --Restore All --TargetPath test`

コマンド	説明
	全ての隔離されたオブジェクトを、**drweb-ctl**コマンドが起動された現在のディレクトリにある`test`サブディレクトリに復元します。 `--Restore All`*では、追加のオプション* `--TargetPath`*（指定された場合）にはファイルへのパスではなくディレクトリへのパスを指定する必要があります。*

脅威および隔離コマンド用のフォーマット出力

出力フォーマットは、オプション引数`--Format`として指定されたフォーマット文字列を使用して定義されます。フォーマット文字列は引用符で囲んで指定する必要があります。フォーマット文字列には、出力中に特定の情報として表示される特殊なマーカーだけでなく、共通の記号（「そのまま」で表示されるもの）を含めることができます。以下のマーカーが利用可能です。

1. `threats`と`quarantine`コマンドに共通：

マーカー	説明
`%{n}`	新しい文字列
`%{t}`	集計
`%{threat_name}`	Doctor Webの分類に従って検出された脅威（ウイルス）の名前
`%{threat_type}`	Doctor Webの分類に従った脅威のタイプ（「既知のウイルス」など）
`%{size}`	元のファイルサイズ
`%{origin}`	パスを含む元のファイルのフルネーム
`%{path}`	`%{origin}`の同義語
`%{ctime}`	元のファイルが変更された日時（*「%Y-%b-%d %H:%M:%S」*フォーマット、例：`2018-Jul-20 15:58:01`）
`%{timestamp}`	`%{ctime}`と似ているが、*UNIXのタイムスタンプ*フォーマット
`%{owner}`	元のファイル所有者のユーザー名
`%{rowner}`	元のファイルのリモートユーザー所有者（該当しない場合や値が不明な場合は？と置き換えられます）

2. `threats`コマンドに固有：

マーカー	説明
`%{hid}`	脅威に関連付けられているイベントの履歴にある脅威レコードのID
`%{tid}`	脅威のID
`%{htime}`	脅威に関連したイベントの日時

マーカー	説明
%{app}	脅威を処理したDr.Web for UNIX Internet GatewaysコンポーネントのID
%{event}	脅威に関連する最新イベント: • FOUND - 脅威が検出されました。 • Cure - 脅威は修復されました。 • Quarantine - 脅威のあるファイルが隔離されました。 • Delete - 脅威のあるファイルが削除されました。 • Ignore - 脅威は無視されました。 • RECAPTURED - 他のコンポーネントによって脅威が再度検出されました。
%{err}	エラーメッセージテキスト(エラーが空の文字列に置き換えられない場合)

3. quarantineコマンドに固有:

マーカー	説明
%{qid}	隔離されたオブジェクトのID
%{qtime}	オブジェクトを隔離に移動した日時
%{curetime}	隔離に移されたオブジェクトの修復を試みた日時(該当しない場合または値が不明の場合は?に置き換えられます)
%{cureres}	隔離されたオブジェクトの修復を試みた結果: • cured - 脅威は修復されています。 • not cured - 脅威は修復されていないか、修復が試みられていません。

例

```
$ drweb-ctl quarantine --Format "{%{n} %{origin}: %{threat_name} - %{qtime}%{n}}"
```

このコマンドは、次のタイプのレコードとして隔離内容を表示します。

```
{
  <path to file>:  <threat name>  -  <date of moving to quarantine>
}
...
```

3.5.情報に関するコマンド

情報に関するコマンドには以下のものがあります。

コマンド	説明
`appinfo`	**目的**：アクティブなDr.Web for UNIX Internet Gatewaysコンポーネントに関する情報を出力します。 現在実行中の各コンポーネントについて、次の情報が表示されます。 • 内部で使用される名前 • プロセスID**GNU/Linux**（PID） • 状態（実行中、停止など） • コンポーネントの動作がエラーによって終了した場合、エラーコード • 追加情報（任意） 設定デーモン（Dr.Web ConfigD）については、以下の追加情報が表示されます。 • インストールされたコンポーネントのリスト—*Installed* • 設定デーモンによって起動する必要のあるコンポーネントのリスト—*Should run* **引数**： None **オプション**： `-f [--Follow]` - コンポーネントのステータス変更に関する新しいメッセージを待ち、それらを受け取り次第メッセージを出力します（Ctrl + Cキーを押して待機を中断します）。
`baseinfo`	**目的**：Virus-Finding Engineの現在のバージョン、およびウイルスデータベースのステータスに関する情報を表示します。 以下の情報が表示されます。 • アンチウイルスエンジンのバージョン • 現在使用されているウイルスデータベースがリリースされた日時 • 使用可能なウイルスレコードの数（ウイルスデータベース内の） • ウイルスデータベースおよびアンチウイルスエンジンが最後に更新された時間 • スケジュールされている次回の自動更新の時間 **引数**： None **オプション**： `-l [--List]` - ダウンロードされたウイルスデータベースのファイルと各ファイルのウイルスレコード数の全リストを表示します。
`certificate`	**目的**：Dr.Web for UNIX Internet Gatewaysによって使用されるDr.Webの信頼できる証明書の内容を表示します。証明書を*<cert_name>*.pemファイルに保存するには、次のコマンドを使用できます。 `$ drweb-ctl certificate > <cert_name>.pem` **引数**： None

コマンド	説明
	オプション: None
events	機能: Dr.Web for UNIX Internet Gatewaysイベントを表示します。そのほか、このコマンドを使用してイベントを管理(既読としてマーク、削除)することができます。 このコマンドはすべてのイベントを「既読」としてマークします。 オプション: --Report *<type>* - イベントレポートのタイプを指定します。 **使用可能な値:** • BRIEF - 短いレポート • DEBUG - 詳細なレポート • JSON - JSON形式のシリアル化されたレポート -f [--Follow] - 新しいイベントを待ち、発生時にそれらを表示します(CTRL + Cはスタンバイを中断します)。 -s [--Since] <date, time> - 指定されたタイムスタンプの前に発生したイベントを表示します(<date, time> は YYYY-MM-DD hh:mm:ss のフォーマットで指定します)。 -u [--Until] <date, time> - 指定されたタイムスタンプの後に発生したイベントを表示します(<date, time> は YYYY-MM-DD hh:mm:ss のフォーマットで指定します)。 -t [--Types] <type list> - 指定されたタイプのイベントのみを表示します(コンマで区切られます)。 次のイベントタイプを使用できます。 ・Mail - メール内で脅威を検出 ・UnexpectedAppTermination - コンポーネントの予期しないシャットダウン すべてのタイプのイベントを表示するには、All を使用します。 --ShowSeen - 既読イベントも表示されます。 --Show *<list of events>* - リストアップされたイベントを表示します(イベント識別子はコンマで区切られます)。 --Delete *<list of events>* - リストアップされたイベントを削除します(イベント識別子はコンマで区切られます)。 --MarkAsSeen *<list of events>* - リストアップされたイベントを既読としてマークします(イベント識別子はコンマで区切られます)。 すべてのイベントを「既読」としてマークする場合や削除する場合は、<events list> ではなく All を指定します。例: `$ drweb-ctl --MarkAsSeen --All`
report	機能: Dr.Web for UNIX Internet Gateways イベントに関するレポートをHTML形式で作成します(ページ本文は指定したファイルに出力されます)。 引数:

コマンド	説明
	<type> - レポートを作成するイベントのタイプです（タイプを1つ指定します）。可能な値については、上記 events コマンドの --Types オプションの説明を参照してください。この引数は必須です。 オプション： -o [--Output] *<path to file>* - 指定したファイルにレポートを保存します。このオプションは必須です。 -s [--Since] *<date, time>* - 指定されたタイムスタンプよりも後に発生したイベントのレポートを作成します（*<date, time>* は YYYY-MM-DD hh:mm:ss のフォーマットで指定します）。 -u [--Until] *<date, time>* - 指定されたタイムスタンプよりも前に発生したイベントのレポートを作成します（<date, time> は YYYY-MM-DD hh:mm:ss のフォーマットで指定します）。 --TemplateDir *<path to directory>* - HTMLレポートテンプレートを含むディレクトリへのパスです。 -s、-u、--TemplateDirは必須のオプションではありません。 `$ drweb-ctl --report Mail -o report.html` たとえば、上記のコマンドは、メールメッセージでのすべての脅威検出イベントに関するレポートをデフォルトのテンプレートで生成し、結果をカレントディレクトリの report.html ファイルに保存します。
idpass *<identifier>*	**目的**：指定されたIDを持つメールメッセージに対する、Dr.Web MailDによって生成され、メールメッセージから削除された脅威を含むアーカイブの保護に使用された、パスワードを表示します（つまり、**RepackPassword**パラメータが"HMAC（*<secret>*）"に設定されている場合のユニークなパスワード）。 **引数**： • *<identifier>* - メールメッセージのID。 **オプション**： -s [--Secret] *<secret>* - アーカイブパスワードの生成に使用される秘密のワード。 *コマンド呼び出し時に秘密のワードが示されていない場合は、現在の秘密のワード<secret>が使用されます。Dr.Web MailD設定に示されます。また、**RepackPassword**パラメータが利用できない場合、またはHMAC（<secret>）と異なる値に設定されている場合、コマンドはエラーを返します。*
license	**目的：現在有効化されているライセンスに関する情報を表示、あるいはデモバージョンのライセンスを取得、あるいはすでに登録されているライセンス（すでにWebサイト上で登録されているものなど）のキーファイルを取得します。** オプションが指定されていない場合は以下の情報が表示されます（スタンドアローンモードのライセンスを使用している場合）： • ライセンス番号 • ライセンスの有効期限が切れる日時

コマンド	説明
	集中管理サーバーから受け取ったライセンスを使用している場合（集中管理モードまたはモバイルモードで製品を使用するため）、以下の情報が表示されます。 **引数**： None **オプション**： `--GetRegistered` *<serial number>* - 新しいキーファイルの提供に関する条件に違反（ライセンスが集中管理サーバーによって管理される場合に製品を集中管理モードで使用していないなど）していない場合、指定されたシリアル番号に対するライセンスキーファイルを取得します。 *シリアル番号が試用期間用のものではない場合、まずDoctor WebのWebサイトでそれを登録する必要があります。* Dr.Web製品のライセンスに関する詳細についてはライセンスセクションをご確認ください。 シリアル番号を取得するには、インターネット接続が必要です。
`log`	**目的**：最後のプログラム複合ログレコードを（*stdout*ストリームで）コンソール画面に表示します（**tail**コマンドと同様）。 **引数**： None **オプション**： `-s [--Size]` *<number>* - 表示される最後のログレコードの数。定義値が0の場合、全てのレコードが表示されます。引数が定義されていない場合は、最後の10件のレコードが表示されます。 `-c [--Components]` *<components list>* - 表示する必要があるレコードのコンポーネントID。IDはコンマ区切りで定義されます。引数が定義されていない場合、全てのコンポーネントによって作成された使用可能なレコードが全て表示されます。 *現在のID（ログに表示される内部コンポーネント名など）は、*のDr.Web for UNIX Internet Gatewaysの構成*のセクションに記載されています。また、*`appinfo`*コマンドを使ってそれらを取得できます（上を参照）。* `-f [--Follow]` - ログで新しいメッセージを待ち、それらを受け取り次第メッセージを表示します（Ctrl + Cキーを押して待機を中断します）。
`stat`	**目的**：ファイルを処理するコンポーネントの操作（CTRL + CまたはQを押すと統計表示が中断されます）、またはネットワークデータスキャンエージェントDr.Web Network Checkerの操作に関する統計を出力します。 出力される統計情報には次が含まれます。 • スキャンを開始したコンポーネントの名前 • コンポーネントのPID • 最後の1分/5分/15分間に処理された1秒あたりのファイルの平均数 • スキャンしたファイルのキャッシュの使用率。 • 1秒あたりの平均スキャンエラー数。 分散されたスキャンエージェントの場合、次の情報が出力されます。

コマンド	説明
	• スキャンを開始したローカルコンポーネントのリスト • スキャン用のファイルを受信したリモートホストのリスト • スキャン用のファイルを送信したリモートホストのリスト 分散されたスキャンエージェントのローカルクライアントの場合、そのPIDと名前が指定されます。リモートクライアントの場合は、ホストのアドレスとポートが指定されます。 ローカルとリモートの両方のクライアントに対して、次の情報が出力されます。 • 1秒間にスキャンされたファイルの平均数 • 1秒あたりの平均送受信バイト数 • 1秒あたりの平均エラー数 **引数：** None **オプション：** `-n [--netcheck]` - ネットワークデータスキャンエージェントの動作に関する統計情報。

使用例

このセクションでは、Dr.Web Ctl（**drweb-ctl**）ユーティリティの使用例を示します。

- オブジェクトのスキャン：
  - シンプルなスキャンのコマンド
  - 条件によって選択されたファイルのスキャン
  - 追加のオブジェクトのスキャン
- 設定の管理
- 脅威の管理
- 自律コピーモードでの動作例

1. オブジェクトのスキャン

1.1. シンプルなスキャンのコマンド

1. デフォルトのパラメータで `/home` ディレクトリのスキャンを実行する：

```
$ drweb-ctl scan /home
```

2. `daily_scan` ファイルに含まれているパスをスキャンする（1行につき1つのパス）：

```
$ drweb-ctl scan --stdin < daily_scan
```

3. *sda*ドライブ上のブートレコードのスキャンを実行する：

```
$ drweb-ctl bootscan /dev/sda
```

4. 実行中のプロセスのスキャンを実行する：

```
$ drweb-ctl procscan
```

1.2. 条件によって選択されたファイルのスキャン

以下は、スキャンの対象となるファイルの選択と、ユーティリティ**find**の操作結果の使用例です。取得したファイルのリストは、パラメータ`--stdin`または`--stdin0`を指定して**drweb-ctl** `scan`コマンドに送信されます。

1. **find** ユーティリティによって返されたリストに含まれ、NUL(`\0`)記号で区切られたファイルをスキャンする:

```
$ find -print0 | drweb-ctl scan --stdin0
```

2. ファイルシステムの1つのパーティション上の、ルートディレクトリから始まり、すべてのディレクトリ内にあるすべてのファイルをスキャンする:

```
$ find / -xdev -type f | drweb-ctl scan --stdin
```

3. `/var/log/messages` および `/var/log/syslog` ファイルを除いて、ルートディレクトリから始まり、すべてのディレクトリ内にあるすべてのファイルをスキャンする:

```
$ find / -type f ! -path /var/log/messages ! -path /var/log/syslog |
drweb-ctl scan --stdin
```

4. ルートディレクトリから始まり、すべてのディレクトリ内にある *root* ユーザーのすべてのファイルをスキャンする:

```
$ find / -type f -user root | drweb-ctl scan --stdin
```

5. ルートディレクトリから始まり、すべてのディレクトリ内にある *root* および *admin* ユーザーのすべてのファイルをスキャンする:

```
$ find / -type f \( -user root -o -user admin \) | drweb-ctl scan --stdin
```

6. ルートディレクトリから始まり、すべてのディレクトリ内にある、UIDが1000～1005の範囲内にあるユーザーのファイルをスキャンする:

```
$ find / -type f -uid +999 -uid -1006 | drweb-ctl scan --stdin
```

7. ルートディレクトリから始まり、ネスティングレベルが5以下のすべてのディレクトリ内にあるファイルをスキャンする:

```
$ find / -maxdepth 5 -type f | drweb-ctl scan --stdin
```

8. サブディレクトリ内のファイルを無視して、ルートディレクトリ内にあるファイルをスキャンする:

```
$ find / -maxdepth 1 -type f | drweb-ctl scan --stdin
```

9. ルートディレクトリから始まり、すべてのディレクトリ内にあるファイルとすべてのシンボリックリンクをスキャンする:

```
$ find -L / -type f | drweb-ctl scan --stdin
```

10. ルートディレクトリから始まり、すべてのディレクトリ内にあるファイルをシンボリックリンクをたどらずにスキャンする:

```
$ find -P / -type f | drweb-ctl scan --stdin
```

11. ルートディレクトリから始まり、すべてのディレクトリ内にある2017年5月1日以前に作成されたファイルをスキャンする:

```
$ find / -type f -newermt 2017-05-01 | drweb-ctl scan --stdin
```

1.3. 追加のオブジェクトのスキャン

1. リモートホスト *192.168.0.1* 上の `/tmp` ディレクトリ内にあるオブジェクトを、*user* ユーザーとしてパスワード *passw* を使用してSSH経由でそれらに接続することでスキャンする:

```
$ drweb-ctl remotescan 192.168.0.1 /tmp --Login user --Password passw
```

2. 設定の管理

1. 実行中のプロセスに関するものを含む、現在のDr.Web for UNIX Internet Gatewaysパッケージに関する情報を表示する:

```
$ drweb-ctl appinfo
```

2. アクティブな設定の `[Root]` セクションからすべてのパラメータを出力する:

```
$ drweb-ctl cfshow Root
```

3. アクティブな設定の `[LinuxSpider]` セクション内で **Start** パラメータに `No` を設定する(これによりファイルシステムモニター SpIDer Guardが無効になります):

```
# drweb-ctl cfset LinuxSpider.Start No
```

このアクションを実行するには、スーパーユーザー権限が必要です。権限を昇格させるには、以下の例に示すように、**sudo**コマンドを使用できます。

```
$ sudo drweb-ctl cfset LinuxSpider.Start No
```

4. Dr.Web for UNIX Internet Gatewaysのアンチウイルスコンポーネントを強制的にアップデートする:

```
$ drweb-ctl update
```

5. Dr.Web for UNIX Internet Gatewaysのコンポーネント設定を再起動する

```
# drweb-ctl reload
```

このアクションを実行するには、スーパーユーザー権限が必要です。権限を昇格させるには、以下の例に示すように、**sudo**コマンドを使用できます。

```
$ sudo drweb-ctl reload
```

6. サーバー証明書が`/home/user/cscert.pem`ファイルである場合に、Dr.Web for UNIX Internet Gatewaysをホスト*192.168.0.1*で動作している集中管理サーバーに接続する:

```
$ drweb-ctl esconnect 192.168.0.1 --Certificate /home/user/cscert.pem
```

7. Dr.Web for UNIX Internet Gateways を集中管理サーバーから切断する:

```
# drweb-ctl esdisconnect
```

このアクションを実行するには、スーパーユーザー権限が必要です。権限を昇格させるには、以下の例に示すように、**sudo**コマンドを使用できます。

```
$ sudo drweb-ctl esdisconnect
```

8. **drweb-update**と**drweb-configd**によってDr.Web for UNIX Internet Gateways ログ内に作成された最後のログレコードを表示する:

```
# drweb-ctl log -c Update,ConfigD
```

3.脅威の管理

1. 検出された脅威に関する情報を表示します。

```
$ drweb-ctl threats
```

2. 隔離されていない脅威を含むファイルをすべて隔離へ移動します。

```
$ drweb-ctl threats --Quarantine All
```

3. 隔離されたファイルのリストを表示します。

```
$ drweb-ctl quarantine
```

4. 隔離からすべてのファイルを復元します。

```
$ drweb-ctl quarantine --Restore All
```

4. 自律コピーモードでの動作例

1. 自律コピーモードでファイルとプロセスをスキャンし、隔離します。

```
$ drweb-ctl scan /home/user -a --OnKnownVirus=Quarantine
$ drweb-ctl quarantine -a --Delete All
```

最初のコマンドは自律コピーモードで `/home/user` ディレクトリにあるファイルをスキャンします。既知のウイルスが含まれるファイルは隔離に移動されます。2番目のコマンドは隔離コンテンツを(自律コピーモードでも)処理し、すべてのオブジェクトを削除します。

設定パラメータ

コマンドラインから製品を管理するためのDr.Web Ctlツールには、Dr.Web for UNIX Internet Gatewaysの統合設定ファイルにパラメータを含む独自のセクションはありません。設定ファイルの`[Root]`セクションに指定されているパラメータを使用します

Dr.Web Web管理インターフェース

このセクションの内容:

概要

Dr.Web for UNIX Internet GatewaysのWebインターフェースでは、以下の操作が可能です。

1. Dr.Web for UNIX Internet Gatewaysコンポーネントの現在の状態を表示し、一部のコンポーネントを起動または停止します。
2. 更新のステータスを表示し、必要に応じて手動で更新プロセスを開始します。
3. 製品ライセンスのステータスを表示し、必要に応じてライセンスキーをロードします。
4. 検出された脅威のリストを表示し、隔離されたオブジェクトを管理します(Dr.Web File Checkerコンポーネントを使用してローカルファイルシステムで検出された脅威のみが表示されます)。
5. Dr.Web for UNIX Internet Gatewaysに含まれるコンポーネントの設定を編集します。
6. Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続するか、スタンドアロンモードに切り替えます。
7. ローカルファイルのオンデマンドスキャンを開始します(ブラウザで開いたページにファイルをドラッグ&ドロップして実行する機能も含みます)。

Webインターフェースのシステム要件

以下のWebブラウザでは、Webインターフェースが正常に動作することが保証されています。

- **Microsoft Internet Explorer**-バージョン11以降。
- **Mozilla Firefox**-バージョン25以降。
- **Google Chrome**-バージョン30以降。

Webインターフェースにアクセスする

Webインターフェースにアクセスするには、ブラウザのアドレスバーに次のようなアドレスを入力します。

```
https://<host_with_drweb>:<port>/
```

ここで、*<host_with_drweb>*は、Dr.Web for UNIX Internet GatewaysがDr.Web HTTPD Webインターフェースサーバーで動作するホストのIPアドレスまたは名前で、*<port>*は、Dr.Web HTTPDがリッスンしている(このホスト上の)ポートです。ローカルホスト上で動作するDr.Web for UNIX Internet Gatewaysコンポーネ

ントにアクセスするには、IPアドレス`127.0.0.1`または名前`localhost`を使用します。デフォルトでは、*<port>*は`4443`です。

したがって、デフォルトではローカルホストのWebインターフェースにアクセスするために、ブラウザのアドレスバーに次のURLを入力します。

```
https://127.0.0.1:4443/
```

管理サーバーとの接続が確立すると、スタートアップページが開き、認証フォームが表示されます。管理機能にアクセスするには、Dr.Web for UNIX Internet Gatewaysが動作しているホスト上で管理権限を持つユーザーのログインとパスワードを指定して、認証フォームに入力します。

必要に応じて、個人ユーザー証明書を使用してWebインターフェースで認証を提供できます。その手順は下記です:

1. 認証局証明書で署名された個人証明書を作成します。
2. 管理用のWebインターフェースに接続するために使用されるブラウザで、署名済み証明書をユーザー認証証明書としてインポートします。
3. Dr.Web HTTPD設定(パラメータ**AdminSslCA**)で、個人証明書に署名する認証局証明書へのパスを指定します。

Webインターフェースでの許可に個人ユーザー証明書を使用する場合、許可フォームは表示されず、ユーザーは*root*として許可されます。

必要に応じて、の付録E. SSL証明書を生成するセクションを参照してください。

メインメニュー

認証に成功すると表示されるWebインターフェースの左ペインにメインメニューがあります。そのメニューアイテムでは以下の操作を実行できます。

- **メイン** - Dr.Web for UNIX Internet Gatewaysのインストール済みコンポーネントとそのステータスの全リストを表示するメインページを開きます。
- **脅威** - サーバー上で検出されたすべての脅威を表示するページを開きます。このセクションでは、これらの検出された脅威を管理できます(感染したオブジェクトの隔離、検出された悪意のあるオブジェクトの再スキャン、修復、削除など)。
- **設定** - サーバーにインストールされているDr.Web for UNIX Internet Gatewaysのコンポーネント設定のページを開きます。
- **情報** - このWebインターフェースのバージョンとウイルスデータベースの状態に関する簡単な情報を表示するページを開きます。
- **ヘルプ** - Dr.Web for UNIX製品のヘルプ情報を含む新しいブラウザタブを開きます。
- **ファイルをスキャン** - ファイルをすばやくスキャンするためのパネルを表示します。このパネルは、閉じられるまでWebインターフェースの開いているページの上部に表示されます。
- **ログアウト** - 現在のWebインターフェースセッションを終了します(ユーザーの個人証明書による認証には使用できません)。

コンポーネントを管理する

メインページでは、Dr.Web for UNIX Internet Gatewaysに含まれるコンポーネントのリストを表示したり、それらの動作を管理したりすることができます。

リスト上にあるDr.Web for UNIX Internet Gatewaysのコンポーネントは、脅威を監視するメインコンポーネントと、Dr.Web for UNIX Internet Gatewaysが正常に動作するよう全体的に管理するサービスコンポーネントの2つのグループに分けられます。メインコンポーネントのリストは、ページ上部に表形式で表示されます（コンポーネントのリストは、ディストリビューション範囲によって異なります）。コンポーネントごとに、以下の情報が表示されます。

1. **名前**。名前をクリックすると、このコンポーネントの設定を含む設定ページが開きます。
2. **状態**。コンポーネントの状態は、スイッチアイコンとコンポーネントの現在の状態に関するメモによって示されます。スイッチをクリックするだけで、コンポーネントを開始または中断できます。スイッチの状態には次のようなものがあります。

	- コンポーネントが無効になっているため、使用されていません。
	- コンポーネントが有効になっており、正しく動作しています。
	- コンポーネントは有効になっていますが、エラーにより動作していません。

   コンポーネントの動作中にエラーが発生した場合は、コンポーネントの状態に関するメモではなく、エラーメッセージが表示されます。アイコンをクリックすると、発生したエラーに関する詳細情報とエラーを解決するための推奨事項がウィンドウに表示されます。
3. **読み込み**。過去1分間、5分間、15分間に、コンポーネントによって1秒間に処理されたファイルの平均数がそれぞれ表示されます（3つの数字がスラッシュ「/」で区切られて表示されます）。
4. **エラー**。過去1分間、5分間、15分間にコンポーネントで1秒間に検出されたエラーの平均数がそれぞれ表示されます（3つの数字がスラッシュ「/」で区切られて表示されます）。

ツールチップを表示するには、アイコンの上にカーソルを置きます。

メインコンポーネントに関する情報を提供する表の下に、サービスDr.Web for UNIX Internet Gatewaysコンポーネント（スキャンエンジン、ファイルスキャンコンポーネントなど）がタイル表示されます。サービスコンポーネントごとに、その状態と動作統計も表示されます。これらのコンポーネントの設定ページを開くには、必要なコンポーネントの名前をクリックします。通常、これらのコンポーネントは必要に応じて自動的に開始または停止されます。ユーザーによって手動で開始および停止される可能性があるサービスコンポーネントについては、名前および動作統計の他に、コンポーネントを開始および停止するためのスイッチもタイルに表示されます。

ページの下部には、ウイルスデータベースが最新かどうかという情報と、ライセンス情報が表示されます。ウイルスデータベースの強制アップデートを実行するには、**更新**をクリックします。**更新**ボタン（またはライセンスの現在の状態によっては**ライセンスをアクティブ化**ボタン）をクリックすると、Dr.Web for UNIX Internet Gatewaysに対して有効なキーファイルをライセンスサーバーにアップロードしてライセンスを更新または有効化できます。

脅威の管理

脅威ページで、検出された脅威の一覧を表示し、それらに対する対応を管理できます。

このページには、ファイルシステムを監視およびスキャンするDr.Web for UNIX Internet Gatewaysのコンポーネント（ネットワークトラフィックをスキャンするコンポーネントは除く）によって検出された脅威のすべてのリストが含まれています。ページ上部には、カテゴリー別に脅威にフィルターを適用できるメニューがあります。

- **ライセンスを有効化** - 検出されたすべての脅威（アクティブな脅威と隔離された脅威の両方を含む）を表示します。
- **アクティブ** - アクティブな（検出されたがまだ駆除されていない）脅威のみを表示します。
- **ブロック済** - ブロックされているすべての脅威、つまり中和されていないが、それを含む感染オブジェクトがブロックされている脅威をすべて表示します。
- **隔離済** - 隔離に移された脅威を表示します。
- **エラー** - エラーのために処理されなかった脅威を示します。

上部メニューの脅威カテゴリーの名称の隣（右側）には、このカテゴリーに分類される検出済みの脅威の数が表示されます。そのカテゴリーに属する脅威が現在表示され、選択されているカテゴリーは、暗い色のフォントで強調されています。必要なカテゴリーの脅威を表示するには、メニューのカテゴリーの名前をクリックしてください。

ネットワークトラフィックをスキャンするコンポーネント（SpIDer Gate、Dr.Web ICAPD）、Dr.Web ClamDによって検出された脅威は、**脅威**ページには表示されません。これらのコンポーネントによって検出された脅威を追跡するために、SNMPを介して使用可能な脅威カウンターと追跡通知を制御できます（Dr.Web SNMPDはMIB Dr.Web構造に従って脅威カウンターと通知へのアクセスを与えます）。

脅威ごとに次の情報が一覧表示されています。

- **ファイル** - 悪意のあるオブジェクトを含むファイルの名前（ファイルパスは指定されていません）。
- **所有者** - 感染ファイルを所有するユーザーの名前。
- **コンポーネント** - 脅威を検出したDr.Web for UNIX Internet Gatewaysのコンポーネントの名前。
- **脅威** -（Doctor Web社の使用する分類に従い）ファイルで検出された脅威の名前。

リストで選択されているオブジェクトについては、以下の情報が表示されます。

- 脅威の名前（Dr.Webウイルス情報ライブラリのページを開くと、その脅威の説明が表示されます）。
- ファイルのサイズ、Byte単位。
- 脅威を検出したコンポーネントの名前。
- 脅威が検出された日時。
- 脅威が最後に変更された日時。
- 感染ファイルを所有しているユーザーの名前。
- ファイルの所有者を含むグループの名前。
- 脅威を含む隔離ファイルに割り当てられた識別子（ファイルが隔離された場合）。
- ファイルの元の場所（脅威の検出時にファイルが存在していた場所）を指すフルパス。

リスト内の任意のオブジェクトをクリックして選択できます。複数のオブジェクトを選択するには、対応するオブジェクトのチェックボックスを選択します。すべてのオブジェクトを選択するか選択をキャンセルするには、脅威リストのヘッダーの**ファイル**フィールドのチェックボックスをオンにします。

リストで選択したオブジェクトにアクションを適用するには、脅威リストの真上にあるツールバーの対応するボタンをクリックします。ツールバーには、次のボタンがあります（選択した脅威の種類によっては、使用できないボタンがあります）。

	-選択されたファイルを削除する（すなわち、永久に削除する）ように指示します。
	-選択したファイルを隔離領域から元の場所に復元するように指示します。
	-選択したファイルに追加のアクションを適用するように指示します（使用可能な操作はドロップダウンリストで指定します）。 • **隔離** - 脅威を含む選択したファイルを隔離するように指示します • **修復** - 脅威を修復しようとします • **無視** - 選択したファイルで検出された脅威を無視し、リストから脅威を削除するように指示します

検索クエリーに基づいて表示された脅威にフィルターを適用することもできます。フィルターを適用して不要な脅威を除外してクエリーに対応するものだけを表示するには、検索ボックスを使用します。検索ボックスはツールバーの右側にマークと一緒に表示されています。脅威リストにフィルターを適用するには、検索ボックスに単語を入力します。名前や説明に入力した単語が含まれていない脅威はすべて非表示になります（このフィルターの適用では大文字と小文字が区別されません）。検索結果を消去してフィルターを適用していないリストを表示するには、検索ボックスのをクリックするか、単語を消去します。

設定を管理する

Dr.Web for UNIX Internet Gatewaysに含まれ、メインページに一覧表示されているコンポーネントの現在の設定パラメータを表示、変更できます。そのためには、**設定**ページを開きます。このページでは、Dr.Web for UNIX Internet Gatewaysを*集中管理*モードまたは*スタンドアロン*モードに切り替えることもできます（これらのモードの詳細については、管理者マニュアルの動作モードを参照してください）。

ページの左側に表示されるメニューには、設定を表示、調整できるすべてのDr.Web for UNIX Internet Gatewaysのコンポーネント名が含まれています。任意のコンポーネントの設定を表示、調整するには、まずこのメニューで該当するコンポーネント名をクリックします。現在、設定の表示および編集を行っているコンポーネントの名前は左側のメニューに強調表示されます。

- メニューの**集中管理**の項目を選択すると、集中管理モードを管理するページに移動します。
- メニューの**全般設定**の項目は、Dr.Web ConfigDコンポーネントの設定に対応しています。これは、Dr.Web for UNIX Internet Gatewaysの全体的な機能を担っています。

コンポーネントにメイン設定のセクションとは別の追加設定のセクションがある場合（たとえば、そのようなセクションはDr.Web ClamDコンポーネントで使用可能で、**ClamAV®**アンチウイルスのインターフェースをエミュレートしてこれらの追加セクションを使用し、異なる接続アドレスを使用するクライアントごとに個別のスキャンパラメータを保持します）、追加のセクションを展開したり折りたたんだりできることを示すアイコンがコンポーネント名の左側に表示されます。アイコンがの場合、追加のセクションは非表示になります。アイコンがの場合は、追加のセクションが1行に1つずつメニューに表示されます。追加セクションのリストを展開したり折りたたんだりするには、必要なコンポーネントの名前の横にある展開・折りたたみアイコンをクリックします。

- 設定を含む追加のセクションは、インデントラインとして表示されます。追加セクションのパラメータを表示または編集するには、その名前をクリックします。

- コンポーネントの設定を含む下位セクションの追加が許可されている場合、コンポーネント名の右側にある＋をクリックして追加します。次に、新しいサブセクションに一意の名前（タグ）を指定して、**OK**をクリックします。サブセクションを作成せずにウィンドウを閉じるには、**キャンセル**をクリックします。
- コンポーネントのサブセクションを削除するには、コンポーネント名にカーソルを合わせると表示されるサブセクション名（タグ）の右側の ✕ をクリックします。次に、サブセクションを削除することを確認して**はい**をクリックするか、**いいえ**をクリックしてサブセクションを削除せずにウィンドウを閉じます。

設定ページの上部には、表示モードを変更できるメニューがあります。以下のモードが利用可能です。

- **ライセンスを有効化** - 表示および調整可能なすべてのコンポーネントの設定パラメータを含むテーブルを表示します。
- **変更** - デフォルトとは異なる値を持つコンポーネントの設定パラメータを含むテーブルを表示します。
- **Ini エディタ** - デフォルト設定とは異なる値を持つこのコンポーネントの設定パラメータを使用してテキストエディターを表示します。表示されるテキストは、設定ファイルと同じ形式です（パラメータ= 値のペアを含みます）。

検索クエリーに基づいて表示されたパラメータにフィルターを適用することもできます。フィルターを適用して不要なパラメータを除外し、クエリーに対応するものだけを表示するには、検索ボックスを使用します。検索ボックスは表示モードメニューの右側に🔍マークと一緒に表示されています。パラメータリストにフィルターを適用するには、検索ボックスに任意の単語を入力します。説明に入力した単語が含まれていないパラメータはすべて非表示になります（このフィルターの適用では大文字と小文字が区別されません）。検索結果を消去してフィルターを適用していないリストを表示するには、検索ボックスの⊗をクリックするか、その中の単語を消去します。

パラメータが表形式で表示されている場合（つまり、**ライセンスを有効化**表示モードと**変更**表示モード）にのみ、フィルターを適用してパラメータを除外できます。

表形式でコンポーネント設定を表示、編集する

表形式でパラメータを表示する場合（**ライセンスを有効化**表示モードと**変更**表示モード）、各表の行にはパラメータの名前と説明（左側）およびその現在値（右側）が含まれます。Boolean値パラメータ（使用可能な値が2つのみの場合、「はい」と「いいえ」）では、値の代わりにチェックボックスが表示されます（チェックを入れると「はい」を、チェックしないと「いいえ」を意味します）。

> （変更されたものだけでなく）すべてのパラメータを表示するように選択した場合、変更された（デフォルト以外の）値は太字で示されます。

完全なパラメータリストはグループに分割されます（**全般**、**アドバンス**など）。グループを折りたたむまたは展開するには、その見出し（名前）をクリックします。グループが折りたたまれていて、そのパラメータが表に表示されていない場合は、グループ名の左側に›のアイコンが表示されます。グループが展開されてテーブルにパラメータが表示されると、⌄のアイコンがグループ名の左側に表示されます。

パラメータを調整するには、表内の現在の値をクリックします（Boolean値パラメータの場合は、対応するチェックボックスのチェックマークを設定または削除します）。パラメータに一連の定義済みの値がある場合は、現在の値をクリックした後にすべてドロップダウンリストとして表示されます。パラメータに数値がある場合は、現在の値をクリックした後に編集ボックスが表示されます。必要な値を指定して、ENTERを押してください。以下の図は、パラメータ値を変更する方法の例を示しています（図に示されているコンポーネントのセットは、提供されているものとは異なる可能性があることに注意してください）。パラメータ値に対するすべての変更は、対応するコンポーネントの設定にすぐに適用されます。

図3. 表形式のコンポーネント設定

パラメータがその値として文字列を要求しているか、任意の値のリストを受け付ける場合は、パラメータの現在の値をクリックして編集すると、ポップアップウィンドウが表示されます。パラメータが値のリストを受け付ける場合は、次の図に示すように複数行の編集ボックスに表示されます（1行に1つの値）。一覧表示されている値を編集するには、編集ボックスで必要な行を変更、削除、または追加する必要があります。

図4. 値リストの編集

パラメータの値を編集したら、**保存**をクリックして変更内容を適用し、ウィンドウを閉じます。変更を適用せずにウィンドウを閉じるには、**キャンセル**をクリックするか、ポップアップウィンドウの右上隅にある×のアイコンをクリックします。

テキストエディターでコンポーネントの設定を表示、編集する

Ini エディタモードでパラメータを表示すると、それらは製品の設定ファイルと同じ形式で表示されます（パラメータ＝値のペア）。ここでのパラメータは、設定ファイル（対応するコンポーネントの設定セクション）に直接書き込まれるパラメータの名前です。このモードでは、デフォルト値とは異なる値のパラメータのみが表示されます（つまり、ラ

イセンスを有効化表示モードで値が太字で強調されているパラメータ）。下の図は、このシンプルビューのテキストエディターでパラメータがどのように表示されるかを示しています。

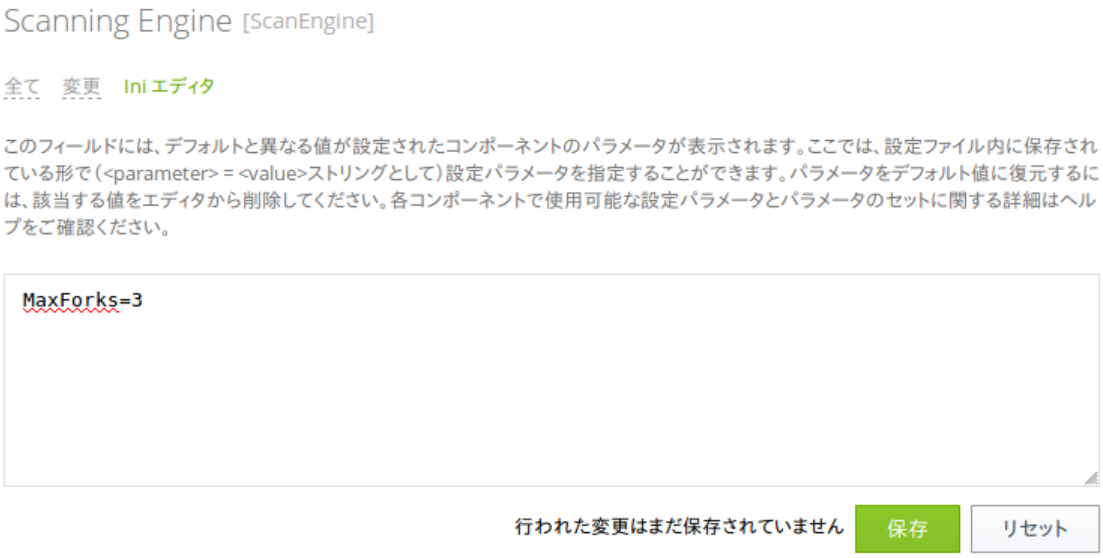

図5. 簡単なテキスト設定エディター

必要な変更を加えるには、設定ファイルの編集について説明したのと同じ規則に従って、このテキストエディターでテキストを編集します（これにより、左側で強調表示されているコンポーネントの設定を含むセクションのみが変更されます）。必要に応じて、コンポーネントで使用可能な任意のパラメータに新しい値を指定できます。この場合、このパラメータの値はデフォルト設定からエディターに入力した値に変わります。パラメータをデフォルト値にリセットする場合は、このテキストエディターでこのパラメータを含む行を消去してください。変更した場合は、変更を保存するとパラメータはデフォルト値に戻ります。

パラメータ値の編集が終了したら、**保存**をクリックして変更を適用するか、**キャンセル**をクリックして変更を破棄します。

保存をクリックすると、テキストが検証されます。プログラムは、すべてのパラメータが存在し、それらの設定値が有効であるかどうかを確認します。エラーが発生した場合は、適切なメッセージが表示されます。

パラメータ値を指定するために重要な設定ファイル、構造、機能の詳細については、「付録D. Dr.Web for UNIX Internet Gateways設定ファイル」のセクションを参照してください。

追加情報

- Dr.Web ConfigDの設定パラメータ（共通設定）。
- Dr.Web ICAPDの設定パラメータ。
- SpIDer Gateの設定パラメータ。
- Dr.Web Firewall for Linuxの設定パラメータ。
- Dr.Web ES Agentの設定パラメータ。
- Dr.Web Updaterの設定パラメータ。
- Dr.Web ClamDの設定パラメータ。

- Dr.Web File Checkerの設定パラメータ。
- Dr.Web Scanning Engineの設定パラメータ。
- Dr.Web Network Checkerの設定パラメータ。
- Dr.Web SNMPDの設定パラメータ。
- Dr.Web CloudDの設定パラメータ。
- Dr.Web LookupDの設定パラメータ。
- Dr.Web StatDの設定パラメータ。
- 集中管理モードの管理。

集中管理モードの管理

Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続したり、スタンドアロンモードに戻して、製品を集中管理サーバーから切断したりすることができます。集中管理を管理できるページを開くには、**設定**ページの設定メニューから**集中管理**という項目を選択します。

Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続したり、集中管理サーバーとの接続を切断したりするには、このページの該当するチェックボックスを使用します。

集中管理サーバーとの接続

集中管理サーバーに接続を試みたときに画面に表示されるポップアップウィンドウでは、集中管理サーバーに接続するためのパラメータを指定する必要があります。

図6. 集中管理サーバーとの接続

ウィンドウ上部のドロップダウンリストから、集中管理サーバーとの接続方法を1つ選択します。3つの方法があります。

- *ファイルから読み込む*
- *手動で設定*
- *自動で検出*

*ファイルから読み込む*オプションを選択した場合は、このウィンドウの該当するフィールドで、接続設定を含むファイルへのパスも指定する必要があります。このファイルはアンチウイルスネットワーク管理者によって提供されます。*手動で設定*オプションを選択した場合は、集中管理サーバーのアドレスとポートを指定する必要があります。*手動で設定*または*自動で検出*オプションでは、サーバーのパブリックキー(ネットワーク管理者またはインターネットサービスプロバイダーによって提供された)を含むファイルへのパスを指定することもできます。

また、集中管理サーバーでの認証用のワークステーション識別子(ID)とパスワードを知っている場合は、**認証(任意)**セクションでそれらを指定できます。これらのフィールドに入力すると、集中管理サーバーへの接続は、正しいIDとパスワードのペアが入力された場合にのみ成功します。これらのフィールドを空白のままにすると、集中管理サーバーへの接続は、集中管理サーバーで承認された場合(自動的またはアンチウイルスネットワーク管理者による承認)にのみ確立します。

さらに、**ワークステーションを「新規端末」として接続**オプションを使用することもできます(新規ユーザーとして接続する場合)。この場合、ワークステーションからの接続に対して集中管理サーバーで新規端末モードが許可されていると、集中管理サーバーはこの接続を承認した後、自動的に一意のIDとパスワードのペアを生成します。今後コンピューターをサーバーに接続する際はこれを使用します。このモードでは、すでにサーバー上にワークステーションの別のアカウントがある場合でも、集中管理サーバーはワークステーション用に新しいアカウントを生成します。

接続パラメータはアンチウイルスネットワーク管理者またはサービスプロバイダーによって提供された指示に厳密に従って指定してください。

集中管理サーバーに接続するには、すべてのパラメータを指定し、**接続**をクリックして、接続が確立されるまで待ちます。サーバー接続を確立せずにウィンドウを閉じるには、**キャンセル**をクリックします。

Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続すると、スタンドアロンモードに戻すまで、その動作は集中管理サーバーによって管理されます。Dr.Web for UNIX Internet Gatewaysが開始されるたびに、集中管理サーバーへの接続が自動的に確立されます。

ローカルファイルをスキャンする

Webインターフェースには、ローカルコンピューター(現在Webインターフェースにアクセスしているコンピューター)に保存されているファイルをスキャンして、ファイルに悪意のあるコンテンツが含まれているかどうか確認する機能があります。スキャンにはDr.Web for UNIX Internet Gatewaysに含まれているスキャンエンジンが使用されます。スキャン対象として選択されたファイルはDr.Web for UNIX Internet Gatewaysが動作しているサーバーに(HTTPプロトコル経由で)アップロードされますが、脅威が見つかった場合でも、スキャン後にファイルがサーバーに保存されることも、隔離されることもありません。スキャンするファイルを送信したユーザーには、スキャンの結果についてのみ通知されます。

この機能は、Dr.Web for UNIX Internet GatewaysディストリビューションにDr.Web Network Checkerコンポーネントが含まれている場合にのみ使用できます。

ローカルファイルをスキャンするパネルを開いて、スキャン用のパラメータを設定する

Webインターフェースのメインメニューで**ファイルをスキャン**項目を選択したときに表示されるローカルファイルのスキャンパネルを使用して、スキャンするファイルを選択してアップロードできます。起動したパネルは、Webインターフェースの右下隅に表示されます。ローカルファイルのスキャンパネルは以下のようになります。

図7. ローカルファイルのスキャンパネル

このパネルを閉じるには、パネルの右上隅にある ✕ をクリックします。⚙ アイコンをクリックすると、ローカルファイルのスキャン設定を表示できます。これには、ファイルをスキャンする最大時間（ローカルコンピューターからサーバーにファイルをアップロードするのにかかる時間は含まれません）、ヒューリスティック解析の使用、圧縮されたオブジェクトの最大圧縮率、コンテナ（アーカイブなど）に圧縮されたオブジェクトの最大ネストレベルなどが含まれます。

図8. ローカルファイルをスキャンするためのパラメータを設定する

変更した設定を適用して、スキャンするファイルを選択できるファイル選択モードに戻るには、**適用**ボタンを押します。設定に変更を適用せずにファイル選択に戻るには、**キャンセル**ボタンを押します。

ローカルファイルのスキャンを開始する

スキャンするファイルを選択してスキャンを開始するには、**スキャンするファイルをここにドラッグするか、またはクリックして選択してくださいファ**と表示されているターゲット領域を左クリックします。そこをクリックすると、各OSのファイルマネージャーの標準的なファイル選択ウィンドウが開きます。スキャン対象として一度に複数のファイルを選択できます。スキャン対象としてディレクトリを選択することはできませんのでご注意ください。ファイルマネージャーウィンドウで選択したファイルをマウスで直接ファイルスキャンパネルのターゲットエリアにドラッグすることもできます。スキャンするファイルを指定すると、Dr.Web for UNIX Internet Gatewaysがインストールされているサーバーへのアップロードが開始されます。ファイルがアップロードされると、スキャンが開始されます。ファイルのアップロードおよびスキャン中に、ファイルスキャンパネルにスキャン手順の全体的な進捗状況が表示されます。

図9. ローカルファイルのスキャンの進捗状況

必要に応じて、**停止**ボタンを押してスキャンを中止できます。スキャンが完了すると、アップロードされたファイルのスキャンに関するレポートがファイルスキャンパネルに表示されます。

図10. スキャンしたローカルファイルの結果

複数のファイルがアップロードされると、スキャンに関する詳細なレポートが利用可能になります。拡張レポートを表示するには、**すべてのファイルについてレポートを見る**というリンクをクリックします。

図11. スキャンしたローカルファイルに関する詳細レポート

レポートを閉じて、パネルでスキャン対象の新しいファイルを選択できる状態に戻るには、**OK**を押します。

ファイルスキャンパネルを閉じているときでも、(スキャンの現在の設定を使用して)ローカルファイルのスキャンを開始することができます。ローカルファイルのアップロードとスキャンを開始するには、ファイルマネージャーウィンドウからブラウザで開いているWebインターフェースのページにドラッグアンドドロップします。

Dr.Web ICAPD

Dr.Web ICAPDコンポーネントは、ICAPプロトコルを介してHTTPプロキシサーバー（**Squid**など）に接続します。通常、HTTPプロキシサーバーは、LANユーザーにインターネットアクセスを提供するために使用されるサーバー（ゲートウェイ）にインストールされます。プロキシサーバーは、Dr.Web ICAPDを外部フィルターとして使用します。したがって、Dr.Web ICAPDは、ユーザーのリクエストとこれらのリクエストに対するサーバーのレスポンスを解析します。外部ネットワーク上にあるリソースへのユーザーアクセスを禁止する必要がある場合、または送信データ（ユーザーリクエストまたはサーバーレスポンス）に脅威が含まれているかエラーのためにスキャンできない場合、Dr.Web ICAPDは、Dr.Web ICAPDによってテンプレートから生成される特別なHTMLページをユーザーに返すようにプロキシサーバーに指示します。

HTTPプロトコルを介して転送されるファイルのスキャンの負荷が高まると、Dr.Web Network Checkerコンポーネントによって利用可能なファイル記述子（ファイルディスクリプタ）が枯渇し、スキャンに問題が生じる可能性があります。この場合、Dr.Web for UNIX Internet Gatewaysに使用可能なファイル記述子の上限を増やす必要があります。

動作原理

Dr.Web ICAPDコンポーネントは、ICAPプロトコル（RFC 3507で説明されている*Internet Content Adaptation Protocol（インターネットコンテンツアダプテーションプロトコル）*）を使用して、Dr.Web for UNIX Internet Gatewaysの外部にあり、LANホストからWebサーバーへのHTTP/HTTPS接続を処理するプロキシサーバーと対話します。

コンポーネントは、ローカルホストからサーバーに送信されたリクエストとサーバーから受信したレスポンスをスキャンする（ICAP用語では「適応する」）タスクをプロキシサーバーから受け取ります。ブラックリストに含まれているURL、または望ましくないWebリソースカテゴリーに属しているURLがクライアントリクエストに含まれている場合、Dr.Web ICAPDはプロキシサーバーに、Webサーバーとの接続を切断し、Dr.Web ICAPDがコンポーネントとともに提供されたテンプレートを使用して生成したHTMLページをクライアントに返すように指示します。このページには、要求されたリソースへのアクセスが拒否されたことをユーザーに通知するメッセージと拒否理由が記載されています。Dr.Web ICAPDがWebサーバーのレスポンスでブロックすべき脅威を検出した場合、同様のページが生成され、プロキシサーバーによってユーザーに返されます。

コンポーネントは、特定のURLがいずれかのカテゴリーに属しているかどうかを確認するために、Doctor Webの更新サーバーから定期的に更新されるWebリソースカテゴリーのデータベースを使用するだけでなく、Dr.Web Cloudサービスも参照します。Doctor Webは、以下のWebリソースカテゴリーを追跡します。

- *InfectionSource* - 悪意のあるソフトウェアを含むWebサイト（「感染源」）。
- *NotRecommended* - アクセスすることが推奨されない不正なWebサイト（「ソーシャルエンジニアリング」を使用しているもの）。
- *AdultContent* - ポルノまたはエロティックなコンテンツ、出会い系サイトなどを含むWebサイト。
- *Violence* - 暴力行為を助長するWebサイトや、さまざまな死亡事故などに関するコンテンツを含むWebサイト。
- *Weapons* - 武器および爆発物に関するWebサイトや、それらの製造に関する情報を提供しているWebサイト。
- *Gambling* - 勝負事、カジノ、オークション、オンラインゲームへのアクセスを提供するWebサイト（賭けサイトを含む）。
- *Drugs* - 麻薬の使用、製造または流通を促進するWebサイト。

- *ObsceneLanguage* - 卑猥な言葉を含む(タイトル、記事などに)Webサイト。
- *Chats* - テキストメッセージのリアルタイム送信を提供するWebサイト。
- *Terrorism* - 攻撃的なプロパガンダ、またはテロ攻撃に関する内容を含むWebサイト。
- *FreeEmail* - メールの無料登録を提供するWebサイト。
- *SocialNetworks* - さまざまなソーシャルネットワークサービス: 一般、仕事、企業、興味、テーマ別出会い系サイト。
- *DueToCopyrightNotice* - 一部の著作物(映画、音楽など)の著作権者によって定義されるWebサイト。
- *OnlineGames* - インターネットへの常時接続を使用してゲームへのアクセスを提供するWebサイト。
- *Anonymizers* - ユーザーが個人情報を隠し、ブロックされたWebリソースにアクセスすることを可能にするWebサイト。
- *CryptocurrencyMiningPool* - 仮想通貨マイニングのための一般的なサービスへのアクセスを提供するWebサイト。
- *Jobs* - 求人検索Webサイト。

設定では、システム管理者は、ユーザーがアクセスすることが望ましくないWebリソースのカテゴリーを指定できます。必要なWebリソースへのアクセスをブロックするために独自のブラックリストを設定し、ユーザーにアクセスを許可するためにホワイトリストを設定することもできます。ホワイトリストに含まれているWebリソースへのアクセスは、それらが望ましくないカテゴリーに属する場合でも許可されます。ローカルブラックリストとWebリソースカテゴリーのローカルデータベースにURLに関する情報がない場合、プログラムはDr.Web Cloudサービスを参照します。これにより、プログラムはURLの悪質性に関する情報が利用できるかどうか確認できます。このような情報は、他のDr.Webの製品からリアルタイムで受信されます。

同一のWebサイトは、複数のカテゴリーに同時に所属させることができます。このようなWebサイトへのユーザーアクセスは、そのWebサイトが属する少なくとも1つのカテゴリーが管理者によって望ましくないと設定されている場合はブロックされます。

Webサイトが管理者によってホワイトリストに含まれる場合でも、データ(Webサイトから送信およびダウンロードされたもの)に脅威が含まれているかどうかがスキャンされます。

ICAPプロトコルの特殊な側面のため、データ(`.iso`イメージ、大きなアーカイブ、ビデオファイルなど)の大部分をスキャンする場合は、長い時間がかかることがあります。スキャンするデータのMIMEタイプに応じて制限を設定することを推奨します。HTTPプロキシサーバーの設定では、ICAPプロトコルによるスキャンのために送信できるデータの最大サイズを制限することも推奨します(プロキシサーバー**Squid**の例を参照)。

Dr.Web Updaterコンポーネントは、Doctor Web更新サーバーからWebリソースカテゴリーのデータベースを定期的かつ自動的に更新するために使用されます。同じコンポーネントは、Dr.Web Scanning Engineスキャンエンジン用のウイルスデータベースを更新するために使用されます。Dr.Web CloudDコンポーネントは、Dr.Web Cloudサービスを参照するために使用されます(クラウドサービスの使用は、Appendixes共通設定で設定され、必要に応じて無効にできます)。転送されたデータをスキャンするために、Dr.Web ICAPDはDr.Web Network Checkerコンポーネントを使用します。後者では、Dr.Web Scanning Engineスキャンエンジンを介してスキャンが開始されます。

HTTPリクエストとレスポンスとをブロックするか、渡すため、Dr.Web ICAPDコンポーネントは組み込みのルールとLuaスクリプトを使用できます。

コマンドライン引数

オペレーティングシステムのコマンドラインからDr.Web ICAPDを起動するには、次のコマンドを使用します。

```
$ <opt_dir>/bin/drweb-icapd [<parameters>]
```

Dr.Web ICAPDは次のパラメータを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-icapd --help
```

このコマンドはDr.Web ICAPDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-icapd`

HTTP Squidプロキシサーバーとの統合

Dr.Web ICAPDとHTTPプロキシサーバー**Squid**の統合については、Squidプロキシサーバーとの統合セクションに記載されています。

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[ICAPD]`セクションで指定されている設定パラメータを使用します。

- コンポーネントパラメータ。
- トラフィックモニタリングとアクセスブロックのルール。

コンポーネントパラメータ

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-icapd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-icapd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-icapd
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。**RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
Start *{Boolean}*	コンポーネントはDr.Web ConfigD設定デーモンによって起動される必要があります。 このパラメータにYes値を指定すると、設定デーモンはただちにコンポーネントを開始するように指示されます。また、No値を指定すると、設定デーモンはただちにコンポーネントを終了するように指示されます。 **デフォルト値**：No
DebugDumpIcap *{Boolean}*	詳細なICAPメッセージをデバッグレベルでログファイルに含めるように指示します（つまり、**LogLevel** = DEBUGを設定した場合）。 **デフォルト値**：No

ListenAddress *{network socket}*	Dr.Web ICAPDがHTTPプロキシサーバーからの接続を待ち受け（リッスン）する必要があるネットワークソケット（IPアドレスとポート）を定義します。 **デフォルト値**：`127.0.0.1:1344`
UsePreview *{Boolean}*	*ICAPプレビュー*モードを使用するようにDr.Web ICAPDに指示します。 *必要でない限り、このパラメータのデフォルト値を変更しないことを推奨します。* **デフォルト値**：`Yes`
Use204 *{Boolean}*	Dr.Web ICAPDが*ICAPプレビュー*モードでないときにもレスポンスコード`204`を返すことを許可するかどうかを定義します。 *必要でない限り、このパラメータのデフォルト値を変更しないことを推奨します。* **デフォルト値**：`Yes`
AllowEarlyResponse *{Boolean}*	Dr.Web ICAPDがICAPの早期レスポンスモードを使用すること、つまりHTTPプロキシサーバーからリクエスト全体を受信する前にクライアントへの「早期」レスポンスの送信を開始することを許可するかどうかを定義します。 *必要でない限り、このパラメータのデフォルト値を変更しないことを推奨します。* **デフォルト値**：`Yes`
TemplatesDir *{path to directory}*	Webリソースをブロックしたときに送信されるHTML通知のテンプレートを含むディレクトリへのパス。 **デフォルト値**：*<var_dir>*`/templates/icapd` • **GNU/Linux**の場合：`/var/opt/drweb.com/templates/icapd` • **FreeBSD**の場合：`/var/drweb.com/templates/icapd`
Whitelist *{domain list}*	*ホワイトリストとして使用できる*ドメインのリスト（つまり、これらのドメインがブロックされたカテゴリーに含まれている場合でも、ユーザーの接続が許可されているドメインのリスト。さらに、このリストに示されているドメインのすべてのサブドメインへユーザーがアクセスすることが許可されます）。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ドメイン`example.com`と`example.net`のリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値

```
[ICAPD]
 Whitelist = "example.com",
"example.net"
```

- 2つの文字列（文字列ごとに1つの値）

```
[ICAPD]
 Whitelist = example.com
 Whitelist = example.net
```

2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。

```
# drweb-ctl cfset ICAPD.Whitelist -a
example.com
# drweb-ctl cfset ICAPD.Whitelist -a
example.net
```

> このパラメータで示されるドメインリストの実際の使用方法は、Dr.Web ICAPDに定義されているWebソースへのアクセス管理ルールでの使用*方法*によって決まります。
>
> デフォルトルールのリスト（下記参照）は、ブロックされるWebソースカテゴリーのリストのドメインがこのリストに含まれている場合でも、このリストのドメイン（およびそのサブドメイン）へのアクセスが提供されることを保証します。さらに、このデフォルトのルールセットは、ホワイトリストドメインからダウンロードされたデータに*脅威が含まれているかどうかがスキャンされる*ことも保証します。

デフォルト値：*（未設定）*

Blacklist

{domain list}

*ブラックリストとして使用できる*ドメインのリスト（つまり、これらのドメインがブロックされたカテゴリーに含まれていない場合でも、ユーザーの接続が禁止されているドメインのリスト。さらに、このリストに示されているドメインのすべてのサブドメインへユーザーがアクセスすることが禁止されます）。

リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。

例：ドメイン`example.com`と`example.net`のリストに追加します。

1. 設定ファイルに値を追加します。

- 1つの文字列に2つの値

```
[ICAPD]
 Blacklist = "example.com",
"example.net"
```

	• 2つの文字列（文字列ごとに1つの値） `[ICAPD]` `Blacklist = example.com` `Blacklist = example.net` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset ICAPD.Blacklist -a example.com` `# drweb-ctl cfset ICAPD.Blacklist -a example.net` ⚠ このパラメータで示されるドメインリストの実際の使用方法は、Dr.Web ICAPDに定義されているWebソースへのアクセス管理ルールでの使用*方法*によって決まります。 デフォルトルールのリスト（以下参照）は、このリストのドメイン（およびそのサブドメイン）へのアクセスが常に禁止されることを保証します。このドメインがホワイトリストとブラックリストに同時に追加される場合、デフォルトのルールにより、そのドメインへのユーザーアクセスは確実にブロックされます。 **デフォルト値**：*（未設定）*
Adlist *{list of strings}*	広告URLを表す正規表現のリスト。ここにリストされているいずれかの正規表現に一致するURLは広告URLと見なされます。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：式「`.*ads.+`」および「`.*/ad/.*\.gif$`」をリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[ICAPD]` `Adlist = ".*ads.+", ".*/ad/.*\.gif$"` • 2つの文字列（文字列ごとに1つの値） `[ICAPD]` `Adlist = .*ads.+` `Adlist = .*/ad/.*\.gif$` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。

	`` # drweb-ctl cfset ICAPD.Adlist -a '.*ads.+' # drweb-ctl cfset ICAPD.Adlist -a '.*/ad/.*\.gif$' ``` *正規表現は、POSIX構文（BRE、ERE）またはPerl構文（PCRE、PCRE2）のいずれかを使用して指定されます。* ⚠ このパラメータで示される式リストの実際の使用方法は、Dr.Web ICAPDに定義されているWebソースへのアクセス管理ルールでの使用*方法*によって決まります。 デフォルトルールのリスト（以下参照）は、このリストのURLへのアクセスが、これらのURLのドメインがホワイトリストにない場合にのみ、常に禁止されることを保証します。 **デフォルト値**：*（未設定）*
BlockInfectionSource *{Boolean}*	悪意のあるソフトウェア（*InfectionSource*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "ICAPD.BlockCategory" :BLOCK as _match ``` **デフォルト値**：`Yes`
BlockNotRecommended *{Boolean}*	非推奨サイト（*NotRecommended*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "ICAPD.BlockCategory" :BLOCK as _match ``` **デフォルト値**：`Yes`
BlockAdultContent *{Boolean}*	アダルトコンテンツ（*AdultContent*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "ICAPD.BlockCategory" :BLOCK as _match ``` **デフォルト値**：`No`

BlockViolence *{Boolean}*	暴力的描写（*Violence*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockWeapons *{Boolean}*	武器に関するWebサイト（*Weapons*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockGambling *{Boolean}*	ギャンブルのWebサイト（*Gambling*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockDrugs *{Boolean}*	麻薬に関するWebサイト（*Drugs*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockObsceneLanguage *{Boolean}*	卑猥な表現（*ObsceneLanguage*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No

BlockChats *{Boolean}*	チャットWebサイト（*Chats*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockTerrorism *{Boolean}*	テロリズムに関するWebサイト（*Terrorism*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockFreeEmail *{Boolean}*	無料メールサービスのWebサイト（*FreeEmail*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockSocialNetworks *{Boolean}*	ソーシャルネットワーキングサイト（*SocialNetworks*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：No
BlockDueToCopyrightNotice *{Boolean}*	著作権者のリクエストに従って追加されたWebサイト（*DueToCopyrightNotice*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**：Yes

BlockOnlineGames *{Boolean}*	オンラインゲームWebサイト(*OnlineGames*カテゴリーに含まれる)への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*(下記の詳細を参照)。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**:`Yes`
BlockAnonymizers *{Boolean}*	アノニマイザーWebサイト(*Anonymizers*カテゴリーに含まれる)への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*(下記の詳細を参照)。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**:`Yes`
BlockCryptocurrencyMiningPools *{Boolean}*	仮想通貨マイニングのための一般的なサービスへのアクセスを提供するWebサイト(*CryptocurrencyMiningPool*カテゴリーに含まれる)への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*(下記の詳細を参照)。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**:`Yes`
BlockJobs *{Boolean}*	求人検索Webサイト(*求人*カテゴリーに含まれます)への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*(下記の詳細を参照)。 `url_category in "ICAPD.BlockCategory" :BLOCK as _match` **デフォルト値**:`No`
ScanTimeout *{time interval}*	Dr.Web ICAPDによって開始された1つのファイルに対するスキャンのタイムアウト。 *1秒から1時間の範囲の値を指定できます。* **デフォルト値**:`30s`
HeuristicAnalysis *{On / Off}*	Dr.Web ICAPDによって開始されたファイルのスキャン中に脅威を検出するためにヒューリスティック解析を使用するかどうかを指定します。ヒューリスティック解析における検出の信頼性は高いのですが、ウイルススキャンに時間がかかります。

	ヒューリスティックアナライザによって検出された脅威に適用されるアクションは、**BlockSuspicious**パラメータ値として指定します。 **使用可能な値**: • `On` - スキャン時にヒューリスティック解析を使用するように指示します。 • `Off` - ヒューリスティック解析を使用しないように指示します。 **デフォルト値**: `On`
PackerMaxLevel *{integer}*	圧縮されたオブジェクトスキャン時の最大ネスティングレベル。Dr.Web ICAPDによって開始されたデータのスキャン中に、より深いネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**: `8`
ArchiveMaxLevel *{integer}*	アーカイブスキャン時の最大ネスティングレベル。Dr.Web ICAPDによって開始されたデータのスキャン中に、より深いネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**: `0`
MailMaxLevel *{integer}*	メールメッセージとメールボックスをスキャンするときの最大ネストレベル。Dr.Web ICAPDによって開始されたデータのスキャン中に、より深いネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**: `0`
ContainerMaxLevel *{integer}*	コンテナ（HTMLページなど）をスキャンするときの最大ネストレベル。Dr.Web ICAPDによって開始されたデータのスキャン中に、より深いネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**: `8`
MaxCompressionRatio *{integer}*	圧縮/パックされたオブジェクトの最大圧縮率（非圧縮サイズと圧縮サイズの比率）。オブジェクトの比率が制限を超えると、そのオブジェクトはDr.Web ICAPDによって開始されたデータのスキャン中にスキップされます。 *圧縮率には2よりも小さい値は指定できません。* **デフォルト値**: `500`

BlockKnownVirus *{Boolean}*	データに既知の脅威が含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：Yes
BlockSuspicious *{Boolean}*	データに未知の脅威（ヒューリスティックアナライザによって検出されたもの）が含まれている場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：Yes
BlockAdware *{Boolean}*	データにアドウェアが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：Yes
BlockDialers *{Boolean}*	データにダイアラープログラムが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：Yes
BlockJokes *{Boolean}*	データにジョークプログラムが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：No

BlockRiskware *{Boolean}*	データにリスクウェアが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります（下記の詳細を参照）。* `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：No
BlockHacktools *{Boolean}*	データにハッキングツールが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります（下記の詳細を参照）。* `threat_category in "ICAPD.BlockThreat" :BLOCK as _match` **デフォルト値**：No
BlockUnchecked *{Boolean}*	データがスキャンできない場合、そのデータの受信または送信をブロックするように指示します。 **デフォルト値**：No
MessageHook *{path to file / Lua function}*	LuaでHTTPメッセージを処理するためのスクリプトまたはそのスクリプトを含むファイルへのパス（LuaでのHTTPメッセージ処理セクションを参照）。 *Lua関数またはファイルパスが指定されていない場合、メッセージはルールに従って処理されます。指定したファイルが利用できない場合、コンポーネントの起動時にエラーが返されます。* **デフォルト値**： *自動的に生成されます。デフォルト設定では、次のようになります。* `local dw = require "drweb"` `local cfg = require "drweb.config"` `local dwl = require "drweb.lookup"` `local rx = require "drweb.regex"` `function message_hook(ctx)` `  if ctx.direction == "request" then` `    local url = ctx.request.url` `    if url.in_list(cfg.blacklist) then` `      return "block"` `    end` `    if not url.in_list(cfg.whitelist) then` `      if rx.search(cfg.adlist, url) or rx.search(cfg.adlist, url.raw) then` `        return "block"` `      end` `      if`

```
url.in_categories(cfg.block_url_categories)
then
           return "block"
       end
     end
   end
   if ctx.body.has_threat{category =
cfg.block_threats} then
     return "block"
   end
   if cfg.block_unchecked and
ctx.body.scan_error then
     return "block"
   end
   return "pass"
end
```

トラフィックモニタリングとアクセスブロックのルール

このセクションには、上記のパラメータの他、7つの*ルールセット*`RuleSet*`（`RuleSet0`、...、`RuleSet6`）も含まれます。これらは、トラフィックスキャン、Webリソースへのユーザーアクセスのブロック、インターネットからのコンテンツのダウンロードのブロックを直接管理します。条件の一部の値（IPアドレス範囲、Webサイトカテゴリーのリスト、Webソースのブラックリストとホワイトリストなど）については、テキストファイルからロードされる値の置き換えがあり、LDAPを介して外部データソースから抽出されます（Dr.Web LookupDコンポーネントを使用）。接続を設定する際には、最終的な解決を含むルールが見つかるまで、すべてのルールが昇順で検査されます。ルールリストのギャップは無視されます。

ルールについては、トラフィックモニタリングのルールの付録D「トラフィックモニタリングのルール」セクションで詳しく説明されています。

ルールを表示して編集する

ルールリストを簡単に編集するために空白が残されています。つまり、ルールを含まない`RuleSet`*<i>*セッがあります（*<i>* - `RuleSet`ルールセット番号）。`RuleSet`*<i>*以外の項目を追加することは*できません*`RuleSet`*<i>*の要素内のルールは追加および削除できます。ルールの表示と編集は、次のいずれかの方法で実行できます。

- （任意のテキストエディターで）設定ファイル設定ファイルを表示、編集する（このファイルにはデフォルトと異なるパラメータのみが保存されます）。
- Web管理インターフェース経由（インストールされている場合）。
- コマンドラインベースのインターフェースを介して - Dr.Web Ctl（**drweb-ctl** `cfshow`および**drweb-ctl** `cfset` コマンド）。

> ルールを編集して設定ファイルを変更した場合は、これらの変更を適用するためにDr.Web for UNIX Internet Gatewaysを再起動します。それには、**drweb-ctl**`reload`コマンドを使用します。

コマンド**drweb-ctl** `cfshow`を使用してルールを表示します。

ルールセット**ICAPD.RuleSet1**のコンテンツを表示するには、このコマンドを使用します。

```
# drweb-ctl cfshow ICAPD.RuleSet1
```

drweb-ctl `cfset`コマンドを使用してルールを編集します（以降、*<rule>* - ルールのテキスト）。

- **ICAPD.RuleSet1**セットのすべてのルールを新しいルールで置き換えます。

  ```
  # drweb-ctl cfset ICAPD.RuleSet1 '<rule>'
  ```

- **ICAPD.RuleSet1**ルールセットに新しいルールを次のように追加します。

  ```
  # drweb-ctl cfset -a ICAPD.RuleSet1 '<rule>'
  ```

- **ICAPD.RuleSet1**セットから特定のルールを次のように削除します。

  ```
  # drweb-ctl cfset -e ICAPD.RuleSet1 '<rule>'
  ```

- **ICAPD.RuleSet1**ルールセットを次のようにデフォルトの状態にリセットします。

  ```
  # drweb-ctl cfset -r ICAPD.RuleSet1
  ```

drweb-ctlツールを使用してルールのリストを編集するときは、追加するルールのテキストを一重引用符または二重引用符で囲み、ルールのテキスト内にある二重引用符の前にはバックスラッシュ（「\」）をエスケープ文字として使用します（ルール自体のテキストに二重引用符が含まれている場合）。

設定の**RuleSet***<i>*変数には、ルールの格納について次のような特徴があります。

- 無条件ルールを追加するときは、条件部分とコロンを省略できます。ただし、そのようなルールは常にルールのリストに文字列「 : *<action>*」として格納されます。
- 複数のアクションを含むルール（「*<condition>* : *<action 1>*, *<action 2>*」など）を追加すると、そのようなルールは基本ルールのチェーン「*<condition>* : *<action 1>*」と「*<condition>* : *<action 2>*」に変更されます。
- ロギングまたはルールは、条件部分で条件の離接（論理和）を許可しないため、論理和を実装するには、各ルールに離接語条件がある条件で一連のルールを記録する必要があります。

接続をスキップするための無条件ルール（`Pass`アクション）を**ICAPDRuleSet1**セットに追加するには、次のコマンドのみを実行します。

```
# drweb-ctl cfset -a ICAPD.RuleSet1 'Pass'
```

ただし、指定したルールセットからこのルールを削除するには、次のコマンドを実行する必要があります。

```
# drweb-ctl cfset -e ICAPD.RuleSet1 ' : Pass'
```

ICAPD.RuleSet1ルールを、接続用の標準テンプレートへのパスを未解決アドレスから変更してブロックを実行するルールセットに追加するには、次のコマンドを実行する必要があります。

```
# drweb-ctl cfset -a ICAPD.RuleSet1 'src_ip not in file("/etc/trusted_ip") :
set http_template_dir = "mytemplates", Block'
```

ただし、このコマンドでは指定したセットに*2つのルール*が追加されるため、ルールのセットからこれらのルールを削除するには、次の2つのコマンドを実行する必要があります。

```
# drweb-ctl cfset -e ICAPD.RuleSet1 'src_ip not in file("/etc/trusted_ip") :
set http_template_dir = "mytemplates"'
# drweb-ctl cfset -e ICAPD.RuleSet1 'src_ip not in file("/etc/trusted_ip") :
Block'
```

「悪意のあるオブジェクト*KnownVirus*や*テロリズム*カテゴリーからのURLが検出された場合はブロックする」などのルールを**ICAPD.RuleSet1**ルールセットに追加するには、このルールセットに次の2つのルールを追加する必要があります。

```
# drweb-ctl cfset -a ICAPD.RuleSet1 'threat_category in (KnownVirus) : Block
as _match'
# drweb-ctl cfset -a ICAPD.RuleSet1 'url_category in (Terrorism) : Block as
_match'
```

ルールのセットからこれらのルールを削除する場合にも、2つのコマンドを実行する必要があります（上の例を参照）。

デフォルトのルールセット

デフォルトでは、次のブロックルールセットが指定されています。

```
RuleSet0 =
RuleSet1 = direction request, url_host in "ICAPD.Blacklist" :BLOCK as
BlackList
RuleSet1 = direction request, url_host not in "ICAPD.Whitelist", url match
"ICAPD.Adlist" :BLOCK as BlackList
RuleSet2 =
RuleSet3 = direction request, url_host not in "ICAPD.Whitelist",
url_category in "ICAPD.BlockCategory" :BLOCK as _match
RuleSet4 =
RuleSet5 = threat_category in "ICAPD.BlockThreat" :BLOCK as _match
RuleSet6 =
```

最初の2つのルールは送信HTTP接続を処理します。接続が試みられるホスト（またはURL）がブラックリストに含まれている場合、その接続はブラックリストに基づいてブロックされ、それ以上のスキャンは実行されません。ホスト（URL）がホワイトリストに含まれておらず、アクセスするのが望ましくないとマークされているWebサイトカテゴリーに属しているか、広告URLを表す正規表現のいずれかに一致する場合、URLは望ましくないカテゴリーに属しているため、接続はブロックされます。

RuleSet5で指定されたルールは、（設定に従って）ブロックする必要がある脅威カテゴリーに属する脅威のHTTPリクエストまたはレスポンスをスキャンします。そのような脅威がある場合、接続は脅威の検出に基づいて

ブロックされます。方向条件が指定されていないため、デフォルトでは、クライアントリクエスト(*request*)とサーバーレスポンス(*response*)の両方が検査されます。

トラフィックモニタリングとアクセスブロックのルールの例

1. IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーに*Chats*以外のすべてのカテゴリーのWebサイトへのアクセスを許可する。

   ```
   src_ip in (10.10.0.0/24), url_category not in (Chats) :PASS
   ```

   ルール

   ```
   url_host in "ICAPD.Blacklist" :BLOCK as BlackList
   ```

   が、指定されたルールより上のルールリストに割り当てられている場合、IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーは、ブラックリストのドメイン、つまりパラメータ`ICAPD.Blacklist`にリストされているドメインへのアクセスもブロックされます。また、このルールが下に割り当てられている場合、IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーは、ブラックリストのWebサイトにもアクセスできます。

   `PASS`の処理は最終的なものであるため、これ以上のルールはチェックされず、したがってダウンロードされたデータのウイルススキャンも実行されません。IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーに、ブラックリストに含まれていない*Chats*以外のすべてのカテゴリーのWebサイトへのアクセスを許可し、同時に脅威のダウンロードをブロックするには、次のルールを使用します。

   ```
   url_category not in (Chats), url_host not in "ICAPD.Blacklist",
   threat_category not in "ICAPD.BlockCategory" :PASS
   ```

2. ビデオファイルのコンテンツ(つまり、MIMEタイプが「`video/*`」のデータ(`*`はMIMEクラス`video`の任意のタイプ))のスキャンを実行しない。

   ```
   content_type in ("video/*") :PASS
   ```

LuaでのHTTPメッセージ処理

このセクションの内容:

概要

Dr.Web ICAPDコンポーネントはLuaプログラムインタプリタを介したインタラクションをサポートします(バージョン5.3.4を使用。Dr.Web for UNIX Internet Gatewaysに同梱)。Luaで記述されたスクリプトは、HTTPプロトコルメッセージの分析と処理のためにコンポーネントで使用できます。

ICAPプロトコルを介してプロキシサーバーからスキャンのために受信したHTTPメッセージ(リクエストまたはレスポンス)の分析は、対応する**MessageHook**パラメータのDr.Web ICAPD設定で指定されたLuaスクリプト(Luaコードまたは必要な処理プログラムを含むファイルへのパスとして指定できます)を使用して実行されます。

メッセージ処理のためのスクリプト

スクリプトの必要条件

ファイルには、メッセージスキャンモジュールのエントリポイントとなるグローバル関数が含まれている必要があります（Dr.Web ICAPDは、新しく受信したメッセージを処理する際にこの関数を呼び出します）。処理関数は、次の呼び出し規則を満たす必要があります。

1. *関数名*は `message_hook`です。
2. *引数*はMessageContextテーブルのみです（関数から処理されたメッセージに関する情報へのアクセス権限を提供します）。
3. *単一の戻り値*は文字列です。戻り値は、スキャンされたメッセージの判定、つまり、メッセージをスキップするかブロックするかを決定します。可能な値：
   - `"pass"`は、受信者に渡されます（サーバーへのHTTPリクエスト、クライアントへのHTTPレスポンス）。
   - `"block"`は、HTTPメッセージは受信者に送信されず、クライアントはブロックされたWebページを含むHTTPレスポンスを受信します。

関数が異なる値を返すか、その実行がエラーのために中断された場合、Dr.Web ICAPDは、スキャンエラーとクライアントへのレスポンスが**BlockUnchecked**設定パラメータの値に依存すると見なします。

スキャンのために受信したすべてのHTTPメッセージについて、Dr.Web ICAPDに*Pass*判定を無条件に返す正しいスクリプト定義の例（以下では、`ctx`引数は*MessageContext*テーブルのインスタンスです）。

```
function message_hook(ctx)
  return "pass"
end
```

例

```
local dwl = require "drweb.lookup"

function message_hook(ctx)

  -- Not to block access to resources at the document website
  -- of Doctor Web
  if ctx.req.url.in_list{"download.geo.drweb.com"} then
      return "pass"
  end

  -- To allow access to users from the WebAdmins group
  -- in Active Directory
  if dwl.check("WebAdmins", "AD@WinRoot", ctx.icap.user) then
      return "pass"
  end

  -- Block access for all the others (to all resources)
  return "block"

end
```

使用中のテーブル

テーブルMessageContext

message_hook関数の入力引数として使用されます。処理されたHTTPメッセージに関する情報（タイプ、ヘッダー、本文、送信者と受信者に関する情報（ある場合））へのアクセスを提供します。テーブルには以下のフィールドを含みます。

フィールド	説明	データタイプ
direction	HTTPメッセージタイプ。次の値を取ります。 • "request" - HTTPリクエスト。 • "response" - HTTPレスポンス。	文字列
icap	ICAPリクエストヘッダーに関する情報。	テーブルICAP
request	HTTPリクエストヘッダーに関する情報。	テーブルRequest
response	HTTPレスポンスヘッダーに関する情報。	テーブルResponse
body	HTTPメッセージ本文の情報。	テーブルBody
無効になったメタメソッド：*なし*		

テーブルICAP

MessageContextテーブルのicapフィールドとして使用されます。HTTPプロキシサーバーからのICAPリクエストに関するデータを格納します。以下のフィールドを含みます。

フィールド	説明	データタイプ
user	ICAPリクエストのX-Client-Usernameヘッダーから取得したユーザーに関する情報。	テーブルUser
src	要求を送信したクライアントのIPアドレス。プロキシサーバーによって送信されたICAP要求のX-Client-IPから取得されます。アドレスが不明な場合はnilになります。	テーブルIpAddress
field	ICAPリクエストヘッダーの配列。	HeaderFieldテーブルの配列
search	正規表現を使用してヘッダーを検索する関数。Perl構文（PCRE）の1つの必須patterns引数（検索パターン：1つ（文字列）または複数（文字列の配列））の正規表現を受け入れます。使用可能なすべてのヘッダーを検索します。引用符で囲まれた文字列を使用する場合は、スラッシュ文字をエスケープする必要があります。 ブール値を返します。 • true - field.name .. ": " .. field.value.decoded文字列が、少なくとも1つのヘッダーに対して指定した少なくとも1つの正規表現と一致する場合。	機能

フィールド	説明	データタイプ
	• `false` - それ以外の場合。	
`value`	1つの必須引数、ヘッダー名（文字列）を受け取る関数。指定された名前を持つ最初のヘッダーの値を返します。この名前のヘッダーがない場合は`nil`を返します。	機能
無効になったメタメソッド: *なし*		

テーブルUser

ユーザーを説明するテーブル。次のフィールドが含まれます（すべてのフィールドはオプションです）。

フィールド	説明	データタイプ
`user`	ドメインなしのユーザー名。	文字列
`domain`	ユーザードメイン。	文字列
無効になったメタメソッド: • `__toString` - この関数は、**User**コンテンツを文字列（UTF-8）として返します。 • `__concat` - この関数は、**User**文字列値と別の文字列を連結します。		

テーブルHeaderField

HTTPまたはICAPメッセージヘッダーを説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
`name`	ヘッダー名。	文字列
`value`	ヘッダー値。	文字列
無効になったメタメソッド: *なし*		

テーブルRequest

HTTPリクエストヘッダーを説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
`method`	リクエストで指定されたHTTPプロトコルメソッド（「POST」など）、ICAPリクエストにHTTPリクエストヘッダーが含まれていない場合は`nil`。	文字列
`url`	HTTPリクエストが送信されるリソースのURL。	テーブルUrl
`content_type`	HTTPリクエストの`Content-Type`ヘッダーから取得される情報。	ContentTypeテーブル
`field`	HTTPリクエストヘッダーの配列。	HeaderFieldテーブルの配列

フィールド	説明	データタイプ
search	正規表現を使用してヘッダーを検索する関数。Perl構文（PCRE）の1つの必須patterns引数（検索パターン：1つ（文字列）または複数（文字列の配列））の正規表現を受け入れます。使用可能なすべてのヘッダーを検索します。引用符で囲まれた文字列を使用する場合は、スラッシュ文字をエスケープする必要があります。 ブール値を返します。 • true - field.name .. ": " .. field.value.decoded文字列が、少なくとも1つのヘッダーに対して指定した少なくとも1つの正規表現と一致する場合。 • false - それ以外の場合。	機能
value	1つの必須引数、ヘッダー名（文字列）を受け取る関数。指定された名前を持つ最初のヘッダーの値を返します。この名前のヘッダーがない場合はnilを返します。	機能
無効になったメタメソッド：*なし*		

テーブルContentType

Content-Typeヘッダーから取得した値を説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
type	メッセージ部分のMIMEタイプ	文字列
subtype	メッセージ部分のサブタイプ	文字列
param	次のフィールドを持つテーブル配列形式のヘッダーパラメータ： • nameはパラメータ名（文字列）です。 • valueはパラメータ値（文字列）です。	テーブル配列
match	1つの必須引数media_types、つまりMIMEタイプを説明する文字列の配列を受け取る関数。リスト内の各文字列は、"type/subtype"、"type/*"、"*/*"のようなものでなければなりません。 ブール値を返します。 • true - 本文のMIMEタイプが指定された文字列（大文字と小文字が区別されない）のいずれかに一致するか、配列に文字列"*/*"が含まれている場合。 • false - それ以外の場合。	機能
無効になったメタメソッド： • __tostringは、デコードされたヘッダー値を返す関数です。 • __concatは、ヘッダーの復号化された値と文字列を結合する関数です。		

テーブルUrl

URLを説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
scheme	スキーム(プロトコル)プレフィックス。例:"http"。ない場合はnil。	文字列
host	ホスト名またはIPアドレス。例:"example.com"。ない場合はnil。	文字列
port	ポート番号。例:80。ない場合、値はnilになります。	番号
path	リソースへのパス。例:"index.html"。ない場合、値はnilになります。	文字列
query	デコードされたリクエストパラメータ。ない場合はnil。	文字列
legal_url	URLがowners_noticeカテゴリーに属する場合、フィールドには所有者のWebサイトへのURLが含まれます。属さない場合は、nilになります。	文字列
in_list	1つの必須引数hosts、つまりホストリスト(文字列の配列)を受け取る関数。ブール値を返します。 • true - hostが指定されたドメインのいずれかのサブドメインであるか、それらのいずれかに一致する場合。 • false - それ以外の場合。	機能
categories	1つのオプション引数filter、つまりUrlCategoryFilterテーブルを受け取る関数(引数がない場合は、空のテーブルを使用するのと同じです)。フィルターで指定されたURL条件を満たすすべてのカテゴリーを反復処理できる反復子関数を返します。	機能
in_categories	1つの必須引数categories、つまりURLカテゴリーリスト(文字列の配列)を受け取る関数。ブール値を返します。 • true - URLが指定されたカテゴリーの少なくとも1つに該当する場合。 • false - それ以外の場合。 categories配列が空の場合は、常にfalseを返します。UrlCategoryFilterテーブルのcategoryフィールドの説明にある可能なカテゴリー値を参照してください。	機能
raw	生のデコードされていないURL。	テーブルRawUrl
無効になったメタメソッド: • __toString - この関数は、**Url**コンテンツを文字列(UTF-8)として返します。 • __concat - この関数は、**Url**文字列値と別の文字列を連結します。		

テーブルRawUrl

復号化されていないURLデータを含むテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
scheme	スキーム（プロトコル）プレフィックス。例：`"http"`。ない場合は`nil`。	文字列
host	ホスト名またはIPアドレス。例：`"example.com"`。ない場合は`nil`。	文字列
port	ポート番号。例：`80`。ない場合、値は`nil`になります。	番号
path	リソースへのパス。例：`"index.html"`。ない場合、値は`nil`になります。	文字列
query	デコードされたリクエストパラメータ。ない場合は`nil`。	文字列
無効になったメタメソッド： • `__toString` - この関数は、**RawUrl**コンテンツを文字列（UTF-8）として返します。 • `__concat` - この関数は、**RawUrl**文字列値と別の文字列を連結します。		

テーブルUrlCategoryFilter

URLカテゴリーのフィルターを説明するテーブル。次のフィールドが含まれます（すべてのフィールドはオプションです）。

フィールド	説明	データタイプ
category	URLが該当するカテゴリーのリスト（大文字と小文字は区別されません）。リストには次の値が含まれる場合があります。 • `"infection_source"` - 感染源。 • `"not_recommended"` - 非推奨のWebサイト。 • `"adult_content"` - アダルトコンテンツ。 • `"violence"` - 暴力。 • `"weapons"` - 武器。 • `"gambling"` - ギャンブル。 • `"drugs"` - 薬物。 • `"obscene_language"` - 卑猥な表現。 • `"chats"` - チャット。 • `"terrorism"` - テロリズム。 • `"free_email"` - 無料メール。 • `"social_networks"` - ソーシャルネットワーク。 • `"owners_notice"` - 著作権者からの申し立てによってリストに登録されたWebサイト。 • `"online_games"` - オンラインゲーム。 • `"anonymizers"` - アノニマイザー。 • `"cryptocurrency_mining_pools"` - 仮想通貨マイニングプール。	文字列または文字列のテーブル

フィールド	説明	データタイプ
	• "jobs" - 求人検索Webサイト。 • "black_list" - ブラックリスト。	
category_not	URLが該当しないカテゴリーのリスト(大文字と小文字は区別されません)。	文字列または文字列のテーブル
無効になったメタメソッド: *なし*		

フィルターフィールドが指定されていない(値がnil)場合、脅威はフィルターと一致します。複数のフィルターフィールドが指定されている場合、条件は接続詞(論理積)によって結合されます。フィルターフィールドがテーブル(リスト)の場合、オブジェクトは少なくとも1つのテーブル(リスト)の項目と一致する必要があります。

テーブルResponse

HTTPレスポンスヘッダーを説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
status	HTTPレスポンスコード、またはICAP要求にHTTPレスポンスヘッダーが含まれていない場合はnil。	番号
reason	レスポンスコードの説明。ない場合はnil。	文字列
content_type	HTTPレスポンスのContent-Typeヘッダーから取得される情報。	テーブルContentType
field	HTTPレスポンスヘッダーの配列。	HeaderFieldテーブルの配列
search	正規表現を使用してヘッダーを検索する関数。Perl構文(PCRE)の1つの必須patterns引数(検索パターン:1つ(文字列)または複数(文字列の配列))の正規表現を受け入れます。使用可能なすべてのヘッダーを検索します。引用符で囲まれた文字列を使用する場合は、スラッシュ文字をエスケープする必要があります。 ブール値を返します。 • true - field.name .. ": " .. field.value.decoded文字列が、少なくとも1つのヘッダーに対して指定した少なくとも1つの正規表現と一致する場合。 • false - それ以外の場合。	機能
value	1つの必須引数、ヘッダー名(文字列)を受け取る関数。指定された名前を持つ最初のヘッダーの値を返します。この名前のヘッダーがない場合はnilを返します。	機能
無効になったメタメソッド: *なし*		

テーブルBody

HTTPメッセージ本文を説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
has_threat	1つのオプション引数filter、つまりThreatFilterテーブルを受け取る関数（引数がない場合は、空のテーブルを使用するのと同じです）。ブール値を返します。 • true - HTTPメッセージ本文に、指定されたfilter条件を満たす脅威が含まれている場合。 • false - それ以外の場合。	機能
threats	1つのオプション引数filter、つまりThreatFilterテーブルを受け取る関数（引数がない場合は、空のテーブルを使用するのと同じです）。HTTPメッセージ本文で検出されたすべての脅威を反復処理できる反復子関数を返します。脅威は、Virusテーブルを使用して説明されます。	機能
content_type	HTTPリクエストまたはレスポンスのContent-Typeヘッダーから取得された本文のMIMEタイプに関する情報が含まれます（分析されるメッセージのタイプによって異なります）。	テーブル ContentType
scan_error	本文スキャンエラーが発生した場合。それ以外の場合はnil。使用可能な値： • "path_not_absolute" - 指定されたパスは絶対パスではありません。 • "file_not_found" - ファイルが見つかりませんでした。 • "file_not_regular" - ファイルは通常のファイルではありません。 • "file_not_block_device" - ブロックデバイスではありません。 • "name_too_long" - 名前が長すぎます。 • "no_access" - アクセスが拒否されました。 • "read_error" - 読み取りエラーが発生しました。 • "write_error" - 書き込みエラー。 • "file_too_large" - ファイルが大きすぎます。 • "file_busy" - ファイルは使用中です。 • "unpacking_error" - アンパックエラー。 • "password_protected" - アーカイブはパスワードで保護されています。 • "arch_crc_error" - CRCアーカイブエラー。 • "arch_invalid_header" - 無効なアーカイブヘッダー。 • "arch_no_memory" - アーカイブを解凍するための十分なメモリがありません。 • "arch_incomplete" - 不完全なアーカイブ。 • "can_not_be_cured" - ファイルを修復できません。	文字列

フィールド	説明	データタイプ
	• "packer_level_limit" - パックされたオブジェクトのネストレベルの上限を超えました。 • "archive_level_limit" - アーカイブのネストレベルの上限を超えました。 • "mail_level_limit" - メールファイルのネストレベルの上限を超えました。 • "container_level_limit" - コンテナのネストレベルの上限を超えました。 • "compression_limit" - 圧縮率の上限を超えました。 • "report_size_limit" - レポートサイズの上限を超えました。 • "scan_timeout" - スキャンタイムアウトの上限を超えました。 • "engine_crash" - スキャンエンジンの障害。 • "engine_hangup" - スキャンエンジンのハングアップ。 • "engine_error" - スキャンエンジンエラー。 • "no_license" - アクティブなライセンスが見つかりません。 • "multiscan_too_late" - マルチスキャンエラー。 • "curing_limit_reached" - 修復試行の上限を超えました。 • "non_supported_disk" - ディスクタイプはサポートされていません。 • "unexpected_error" - 予期しないエラー。	
無効になったメタメソッド: *なし*		

テーブルVirus

脅威を説明するテーブル。以下のフィールドを含みます。

フィールド	説明	データタイプ
name	脅威の種類（Doctor Webの分類による）	文字列
type	脅威の種類（Doctor Webの分類による）。可能な値： • "known_virus" - 既知の脅威（ウイルスデータベースに説明がある脅威）。 • "virus_modification" - 既知の脅威の亜種。 • "unknown_virus" - 未知の脅威、疑わしいオブジェクト。 • "adware" - 広告プログラム。 • "dialer" - ダイアラープログラム • "joke" - ジョークプログラム。 • "riskware" - 潜在的に危険なプログラム。	文字列

フィールド	説明	データタイプ
	• "hacktool" - ハッキングツール。	
無効になったメタメソッド: *なし*		

テーブルThreatFilter

脅威に対するフィルターを説明するテーブル。次のフィールドが含まれます（すべてのフィールドはオプションです）。

フィールド	説明	データタイプ
category	脅威が該当するカテゴリーのリスト（大文字と小文字は区別されません）。Virusテーブルのtypeフィールドの説明にあるカテゴリーのリストを参照してください。	文字列または文字列のテーブル
category_not	脅威が該当しないカテゴリーのリスト（大文字と小文字は区別されません）。	文字列または文字列のテーブル
無効になったメタメソッド: *なし*		

フィルターフィールドが指定されていない（値がnil）場合、脅威はフィルターと一致します。複数のフィルターフィールドが指定されている場合、条件は接続詞（論理積）によって結合されます。フィルターフィールドがテーブル（リスト）の場合、オブジェクトは少なくとも1つのテーブル（リスト）の項目と一致する必要があります。

使用例：

1.HTTPメッセージで見つかったすべての脅威の名前をログに出力するには、次の手順を実行します。

```
local dw = require "drweb"

function message_hook(ctx)
  for virus in ctx.body.threats() do
   dw.notice("threat found: " .. virus.name)
  end
  return "pass"
end
```

2.カテゴリーフィルターに一致する脅威名と、脅威が検出されたメッセージ部分の名前をログに出力するには、次の手順を実行します。

```
local dw = require "drweb"

function message_hook(ctx)
  for v in ctx.body.threats({category = "known_virus"}) do
   dw.notice("found known virus: " .. v.name)
  end
  return "pass"
end
```

利用可能な補助モジュール

LuaのプログラムスペースでDr.Web for UNIX Internet Gatewaysとやり取りするために、次の特定のモジュールをインポートできます。

モジュール名	機能
drweb	Luaプログラムを起動したDr.Web for UNIX Internet GatewaysコンポーネントとLuaプロシージャの非同期実行の手段のログに、Luaプログラムからのメッセージを記録する機能を提供するモジュール。
drweb.lookup	Dr.Web LookupDモジュールを呼び出して外部ソースからデータを要求するためのツールを提供するモジュール。
drweb.regex	文字列と正規表現を一致させるためのインターフェースを提供するモジュール。
drweb.config	Dr.Web ICAPD設定パラメータ値を持つテーブルを提供するモジュール。

drwebモジュールの内容

1.機能

このモジュールは次の機能を提供します。

1.1.Luaプログラムからのメッセージを Dr.Web for UNIX Internet Gatewaysコンポーネントログに保存するための機能:

- `log(`*<level>, <message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*<level>*レベル(必要なレベルは、「*debug*」、「*info*」、「*notice*」、「*warning*」、「*error*」を使用して定義します)で書き込みます。
- `debug(`*<message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*DEBUG*レベルで書き込みます。
- `info(`*<message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*INFO*レベルで書き込みます。
- `notice(`*<message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*NOTICE*レベルで書き込みます。
- `warning(`*<message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*WARNING*レベルで書き込みます。
- `error(`*<message>*`)` は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*ERROR*レベルで書き込みます。

1.2.Luaプロシジャの同期管理のための機能:

- `sleep(`*<sec.>*`)` はこのLuaプロシジャインスタンスの実行を指定された秒数で一時停止します。
- `async(`*<Lua function>*`[,` *<argument list>*`])` は、指定された関数の非同期開始を起動し、指定された引数リストに転送します。`async`関数呼び出しはすぐに完了し、戻り値(**Future**テーブル)を使用すると、*<Lua function>*の結果を取得できます(まだ実行が完了していない場合は、完了するまで待機している可能性があります)。

1.3.IPアドレスをIpAddressテーブルとして表示するための機能:

- `ip(<address>)`は、**IpAddress**テーブルの形式で*<address>*文字列として送信される、IPアドレスを指定します。IPv4またはIPv6アドレスのいずれかを使用できます。

1.4.テキストファイルから外部データをアップロードするには:

- `load_set(<file path>)`は、指定されたテキストファイルのコンテンツから`true`値を含むテーブルを生成し、ファイルから読み取られた文字列はキーとして使用されます。空の文字列と空白文字のみで構成される文字列は無視され、テーブルには含まれません。
- `load_array(<file path>)`は、指定されたテキストファイルのコンテンツから文字列の配列を生成します。空の文字列と空白文字のみで構成される文字列は無視され、配列には含まれません。

2.テーブル

2.1.**Future**テーブルは、`async`関数を使用して関数を実行した後の保留中の結果を表します。テーブルには以下のフィールドを含みます。

フィールド	説明	データタイプ
`wait`	`async`関数を使用して開始した関数の結果を返す関数。関数がまだ実行を完了していない場合は、完了を待って結果を返します。`wait`が呼び出される前に関数が完了した場合、結果はすぐに返されます。開始された関数が失敗した場合、`wait`呼び出しは同じエラーを生成します。	機能
無効になったメタメソッド: *なし*		

2.2 **IpAddress**テーブルはIPアドレスを表します。以下のフィールドを含みます。

フィールド	説明	データタイプ
`belongs`	指定したサブネット(IPアドレス範囲)の**IpAddress**テーブルに保存されているIPアドレスの所属を確認する関数。 「*<IP address>*」または「*<IP address>*/*<mask>*」のような文字列を唯一の引数として受け取ります。ここで、*<IP address>*はホストアドレスまたはネットワークアドレス(「127.0.0.1」など)、*<mask>*はサブネットワークマスク(「255.0.0.0」などのIPアドレスとして、または「8」などの数値形式で指定できます)です。 ブール値を返します。 • `true`は、アドレスが指定されたアドレスの少なくとも1つと等しいか、指定されたサブネット(IPアドレスの範囲)の少なくとも1つに属していることを示します。 • `false` - それ以外の場合。	機能
無効になったメタメソッド: • `__tostring`は、文字列内の**IpAddress**を変更する関数です。例:「127.0.0.1」(IPv4)または「::1」(IPv6) • `__concat`は、**IpAddress**を文字列に結合する関数です。 • `__eq`は、2つの**IpAddress**が等しいことを確認するための関数です。		

フィールド	説明	データタイプ
• __bandは、マスクを適用するための関数（例：dw.ip('192.168.1.2') & dw.ip('255.255.254.0')）です。		

3.例

3.1.非同期的に開始される手順によって生成される、ログへの出力メッセージ:

```
local dw = require "drweb"

-- この関数は2秒間待機して文字列を返します。
-- 引数として受信されます
function out_msg(message)
 dw.sleep(2)
 return message
end

-- 「メイン」関数
function intercept(ctx)
 -- NOTICEレベルでDr.Web for UNIX Internet Gatewaysログに文字列を出力します
 dw.notice("Intercept function started.")

 -- out_msg関数の2つのコピーを非同期で起動します
 local f1 = dw.async(out_msg, "Hello,")
 local f2 = dw.async(out_msg, " world!")

 -- out_msg関数のコピーの完了を待機中です
 -- out_msgとその結果をログに出力します
 -- Dr.Web for UNIX Internet Gateways ログにデバッグレベルで出力します
 dw.log("debug", f1.wait() .. f2.wait())
end
```

3.2.指定されたスケジュールに従って定期的に開始する手順を作成する:

```
local dw = require "drweb"

-- Futureテーブルをfutureグローバル変数で保存し、
-- Luaのガベージコレクターにより
-- 計画されたタスクが破壊されないようにします
future = dw.async(function()
   while true do
     -- 毎日、次のメッセージがログに表示されます
     dw.sleep(60 * 60 * 24)
     dw.notice("A brand new day began")
   end
end)
```

3.3.文字列のIPアドレスを変更する:

```
local dw = require "drweb"

local ipv4 = dw.ip("127.0.0.1")
local ipv6 = dw.ip("::1")
local mapped = dw.ip("::ffff:127.0.0.1")
```

drweb.lookupモジュールの内容

1.機能

このモジュールは次の機能を提供します。

- `lookup(`*<request>*`, `*<parameters>*`)`はDr.Web LookupDモジュールから利用できる外部ストレージからデータを要求します。*<request>*パラメータは、Dr.Web LookupD設定内のセクション(文字列*<type>*`@`*<tag>*)に対応している必要があります。*<parameters>*引数はオプションで、リクエストを生成するために使用される置換を表します。これらのパラメータはテーブルとして設定されます。このテーブルのキーと値は文字列でなければなりません。この関数は、リクエストの結果である文字列の配列を返します。
- `check(`*<checked string>*、*<request>*、*<parameters>*)は、Dr.Web LookupDモジュールを介して利用できる外部リポジトリで*<checked string>*が見つかった場合に`true`を返します。引数*<request>*および*<parameters>*は、`lookup`関数の引数と同じです(上記を参照)。*<checked string>*引数は、文字列または`__tostring`メタメソッドを持つテーブル(つまり、文字列にフォーマットできる)であると想定されます。

2.例

データソース`LookupD.LDAP.users`から取得されたユーザーリストのログへの出力:

```
local dw = require "drweb"
local dwl = require "drweb.lookup"

-- 「メイン」関数
function intercept(ctx)
 -- NOTICEレベルでDr.Web for UNIX Internet Gatewaysログに文字列を出力します
 dw.notice("Intercept function started.")

 -- リクエスト結果をDr.Web for UNIX Internet Gatewaysログへ出力
 -- 'ldap@users'データソースへ
 for _, s in ipairs(dwl.lookup("ldap@users", {user="username"})) do
   dw.notice("Result for request to 'ldap@users': " .. s)
 end

end
```

drweb.regexモジュールの内容

1.機能

このモジュールは次の機能を提供します。

- `search(`*<template>*`, `*<text>*`[, `*<flags>*`])` - *<text>*文字列に*<template>*正規表現と一致するサブストリングが含まれている場合は`true`を返します。オプションの*<flags>*パラメータ(整数)は、関数の動作に影響を与える一連のフラグであり、論理和でつなげられます。

- match(*<template>*, *<text>*[, *<flags>*]) - *<template>*正規表現がそのサブストリングだけでなく*<text>*ストリング全体と一致しなければならない点を除いてsearchと同じです。

2.利用可能なフラグ

- ignore_caseはテキストの大文字と小文字を区別しません。

3.例

```
local rx = require "drweb.regex"

rx.search("te.?t", "some TexT") -- false
rx.search("te.?t", "some TexT", rx.ignore_case) -- true

rx.match("some.+", "some TexT") -- true
```

drweb.configモジュールの内容

1.機能

このモジュールには関数はありません。

2.利用可能なテーブル

このモジュールは、次のフィールドを持つテーブルを提供します。

フィールド	説明	データタイプ
whitelist	**Whitelist**設定パラメータの値。	文字列の配列
blacklist	**Blacklist**設定パラメータの値。	文字列の配列
adlist	**Adlist**設定パラメータの値。	文字列の配列
block_url_categories	ブロックされるURLカテゴリーのリスト(Yesに設定された**Block***パラメータ値に基づく)。	文字列の配列
block_threats	ブロックされる脅威カテゴリーのリスト(Yesに設定された**Block***パラメータ値に基づく)。	文字列の配列
block_unchecked	**BlockUnchecked**設定パラメータの値。	論理
無効になったメタメソッド: *なし*		

3.例

```
local cfg = require "drweb.config"

function message_hook(ctx)

  -- Block messages containing threats
  -- from the list of threats to be blocked
  if ctx.body.has_threat{category = cfg.block_threats} then
    return "block"
  end

  -- To permit access to all other resources
  return "pass"

end
```

SpIDer Gate

このコンポーネントは**GNU/Linux** OSのディストリビューションにのみ含まれています。

ネットワークトラフィックとURLを監視するコンポーネントSpIDer Gateは、データ(ネットワークからローカルコンピューター、ローカルホストからネットワークにダウンロードされたデータ)の脅威を検査し、不要なカテゴリーに含まれるWebリソースや管理者によって定義されたブラックリストに含まれるネットワークホストとの接続を防止します。

コンポーネント設定では、スキャン対象のプロトコルの種類を指定することができます。

コンポーネントは、URLがいずれかのカテゴリー(HTTP/HTTPSプロトコルを利用する接続のスキャンに使用される)に属しているかどうかを確認するために、Doctor Webの更新サーバーから定期的に更新されるWebリソースカテゴリーのデータベースを使用するだけでなく、Dr.Web Cloudサービスも参照します。Doctor Webは、以下のWebリソースカテゴリーを追跡します。

- *InfectionSource* - 悪意のあるソフトウェアを含むWebサイト(「感染源」)。
- *NotRecommended* - アクセスすることが推奨されない不正なWebサイト(「ソーシャルエンジニアリング」を使用しているもの)。
- *AdultContent* - ポルノまたはエロティックなコンテンツ、出会い系サイトなどを含むWebサイト。
- *Violence* - 暴力行為を助長するWebサイトや、さまざまな死亡事故などに関するコンテンツを含むWebサイト。
- *Weapons* - 武器および爆発物に関するWebサイトや、それらの製造に関する情報を提供しているWebサイト。
- *Gambling* - 勝負事、カジノ、オークション、オンラインゲームへのアクセスを提供するWebサイト(賭けサイトを含む)。
- *Drugs* - 麻薬の使用、製造または流通を促進するWebサイト。
- *ObsceneLanguage* - 卑猥な言葉を含む(タイトル、記事などに)Webサイト。
- *Chats* - テキストメッセージのリアルタイム送信を提供するWebサイト。
- *Terrorism* - 攻撃的なプロパガンダ、またはテロ攻撃に関する内容を含むWebサイト。
- *FreeEmail* - メールの無料登録を提供するWebサイト。
- *SocialNetworks* - さまざまなソーシャルネットワークサービス: 一般、仕事、企業、興味、テーマ別出会い系サイト。
- *DueToCopyrightNotice* - 一部の著作物(映画、音楽など)の著作権者によって定義されるWebサイト。
- *OnlineGames* - インターネットへの常時接続を使用してゲームへのアクセスを提供するWebサイト。
- *Anonymizers* - ユーザーが個人情報を隠し、ブロックされたWebリソースにアクセスすることを可能にするWebサイト。
- *CryptocurrencyMiningPool* - 仮想通貨マイニングのための一般的なサービスへのアクセスを提供するWebサイト。
- *Jobs* - 求人検索Webサイト。

システム管理者は、ホストが属するカテゴリーに基づいて、望ましくないホストへのアクセスを設定できます。またユーザーは、特定のホストへのアクセスをブロックするために独自のブラックリストを設定したり、アクセスを許可したりするためにホワイトリストを設定することもできます。ホワイトリストのホストへのアクセスは、それらが望ましくない

カテゴリーに属する場合でも許可されます。ローカルのブラックリストとWebリソースのカテゴリーのデータベースにURLに関する情報がない場合、コンポーネントはDr.Web Cloudサービスを参照して、他のDr.Web製品から受信するこうしたURLに悪意があるかどうかの情報をリアルタイムで検査できます。

同一のWebサイトは、複数のカテゴリーに同時に所属させることができます。そのようなWebサイトへのアクセスは、不要なカテゴリーのいずれに属する場合もブロックされます。

Webサイトがホワイトリストに含まれる場合でも、データ(Webサイトから送信およびダウンロードされたもの)に脅威が含まれているかどうかがスキャンされます。

HTTPプロトコルを介して転送されるファイルのスキャンの負荷が高まると、Dr.Web Network Checkerコンポーネントによって利用可能なファイル記述子(ファイルディスクリプタ)が枯渇し、スキャンに問題が生じる場合があります。この場合、Dr.Web for UNIX Internet Gatewaysに使用可能なファイル記述子の上限を増やす必要があります。

Dr.Web for UNIX Internet Gatewaysは、パブリックアクセスのあるWebサーバー、インターネット、外部ネットワークなどの会社のサーバー間に「壁」を作るために組織内で使用できます。デフォルトで、Dr.Web ICAPDコンポーネントがユーザーアクセスを制御する機能を実行するためです。このコンポーネントは、ローカルネットワークからインターネットへのアクセスを提供するプロキシサーバーと連携して動作します。

動作原理

SpIDer Gateコンポーネントは、ユーザーアプリケーションによって確立されたネットワーク接続を監視します。コンポーネントは、クライアントアプリケーションが接続しようとしているサーバーが、設定で不要と指定されているWebリソースカテゴリーのいずれかに属するかどうかを検査します。さらに、コンポーネントはDr.Web Cloudを参照してURLを検査できます。URLが不要なカテゴリー(Dr.Web Cloudのリクエストによって返されたものを含む)またはシステム管理者によって定義されたブラックリストのいずれかに属する場合、接続は中断され、アクセスが許可されていないというメッセージが含まれるHTMLページが表示されます(HTTP/HTTPS接続の場合)。HTMLページは、コンポーネントに付属するテンプレートに従ってSpIDer Gateによって生成されます。このページには、リクエストされたリソースにアクセスできないという通知とブロックに関する詳細が表示されます。SpIDer Gateがブロックする必要のある脅威を見つけた場合、同様のページが表示され、クライアントに返されます。接続にHTTP(S)とは異なるプロトコルが使用されている場合、コンポーネントはこのサーバーとの接続を確立するための許可のみをスキャンします。

補助コンポーネントDr.Web Firewall for Linuxは、クライアントアプリケーションによって確立されたリモートサーバーとの接続をリダイレクトします。コンポーネントは、**GNU/Linux**システムコンポーネントの**NetFilter**ルールの動的管理を実行します。

Dr.Web for UNIX Internet Gateways内では、クライアントアプリケーションは会社の保護されたサーバーリソースになります(パブリックアクセスのあるWebサーバーなど)。デフォルトで、Dr.Web ICAPDコンポーネントがローカルネットワークユーザーのインターネットへのアクセスを管理する機能を実行するためです。このコンポーネントは、ローカルネットワークからインターネットへのアクセスを提供するプロキシサーバーと連携して動作します。

Dr.Web Updaterコンポーネントは、Doctor Web更新サーバーからWebリソースカテゴリーのデータベースを定期的かつ自動的に更新するために使用されます。同じコンポーネントは、Dr.Web Scanning Engineスキャンエンジン用のウイルスデータベースを更新するために使用されます。Dr.Web CloudDコンポーネントは、Dr.Web Cloudサービスを参照するために使用されます(クラウドサービスの使用は、Appendixes共通設定で設定され、必要に応じて無効にできます)。転送されたデータを検査するために、SpIDer GateはDr.Web Network

Checkerコンポーネントを使用します。後者では、Dr.Web Scanning Engineスキャンエンジンを介してスキャンが開始されます。

コマンドライン引数

SpIDer Gateを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-gated [<parameters>]
```

SpIDer Gateは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-gated --help
```

このコマンドはSpIDer Gateに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-gated`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[GateD]`セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-gated • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-gated • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-gated
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。 **RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 最小値 - 10s. **デフォルト値**：30s
TemplatesDir *{path to directory}*	Webリソースをブロックしたときに送信されるHTML通知のテンプレートを含むディレクトリへのパス。 **デフォルト値**：*<var_dir>*/templates/gated • **GNU/Linux**の場合：/var/opt/drweb.com/templates/gated • **FreeBSD**の場合：/var/drweb.com/templates/gated
CaPath *{path}*	信頼できるルート証明書のシステムリストを含むディレクトリまたはファイルへのパス。 **デフォルト値**：*信頼できる証明書のリストへのパス。パスはお使いの**GNU/Linux**ディストリビューションに依存します。* • **Astra Linux**、**Debian**、**Linux Mint**、**SUSE Linux**、**Ubuntu**の場合、*通常は*/etc/ssl/certs/*です。* • **CentOS**と**Fedora**の場合、/etc/pki/tls/certs/ca-bundle.crt*です。* • *他のディストリビューションでは、コマンド***openssl** version -d*の実行結果によってパスを定義できます。*

	• *コマンドが使用できない場合、またはOSディストリビューションを特定できない場合は、値* `/etc/ssl/certs/` *が使用されます。*

接続スキャンの設定を変更しても、変更を加える前にアプリケーションによってすでに確立されている接続のスキャンには影響しません。

補助コンポーネントDr.Web Firewall for Linuxの設定で、トラフィックモニタリングのより具体的なパラメータを指定します。

Dr.Web Firewall for Linux

このコンポーネントは**GNU/Linux** OSのディストリビューションにのみ含まれています。

コンポーネントを正しく動作させるため、以下のオプションを指定してOSカーネルが構築されている必要があります。

- *CONFIG_NETLINK_DIAG, CONFIG_INET_TCP_DIAG;*
- *CONFIG_NF_CONNTRACK_IPV4, CONFIG_NF_CONNTRACK_IPV6, CONFIG_NF_CONNTRACK_EVENTS*
- *CONFIG_NETFILTER_NETLINK_QUEUE, CONFIG_NETFILTER_NETLINK_QUEUE_CT, CONFIG_NETFILTER_XT_MARK.*

必要なオプションの組み合わせは、使用する **GNU/Linux** のディストリビューションキットによって異なります。

Dr.Web Firewall for Linuxは補助コンポーネントです。SpIDer Gateに対して接続マネージャー機能を実行します。Dr.Web Firewall for Linuxは、ホスト接続がSpIDer Gateを通過し、接続トラフィックが監視されるようにします。

動作原理

このセクションの内容:

概要

Dr.Web Firewall for Linuxコンポーネントにより、SpIDer Gateが正しく動作するようにします。**NetFilter**（**GNU/Linux**OSコンポーネント）用に調整されたルーティングルールを解析し、確立された接続が、クライアントアプリケーションとリモートサーバー間の中間の機能（プロキシ）を実行するSpIDer Gateにリダイレクトされるように修正します。

Dr.Web Firewall for Linuxは、トランジット接続だけでなく、送信と受信のリダイレクトのルールを別々に管理できます。迂回またはリダイレクトのルールを正確に設定するには、コンポーネントではLuaで書かれたスクリプトだけでなく設定に組み込まれたルールを使うことができます。

接続監視のメカニズム

接続を監視するために、Dr.Web Firewall for Linuxはルーティングポリシーのデータベース（**man** `ip`: `ip route`、`ip rule`参照）と**NetFilter**システムコンポーネントの*nf_conntrack*インターフェースで指定されたルーティングテーブルを使用します。監視された接続と転送されたパケットには、Dr.Web Firewall for Linuxが**NetFilter**のチェーンのさまざまな段階で接続をリダイレクトし、転送されたパケットを正しく処理できるようにするビットマークが付けられます（詳細は**man** `iptables`を参照）。

iptablesルールのアクション

Dr.Web Firewall for Linuxは、**iptables**ルールで次のアクションを使用します。

- *MARK*: このアクションにより、Dr.Web Firewall for Linuxは指定された数字マークをパケットに設定できます。
- *CONNMARK*: このアクションにより、Dr.Web Firewall for Linuxは指定された数字マークを接続に設定できます。
- *TPROXY*: このアクションにより、Dr.Web Firewall for Linuxは、パケットのコンテンツを変更することなく、*PREROUTING* **NetFilter**チェーンから指定されたネットワークソケット(*<IP address>*: *<port>*)にパケットをリダイレクトできます。このアクションを使用すると、Dr.Web Firewall for Linuxは接続の最初の宛先アドレスを特定できます。
- *NFQUEUE*このアクションにより、エンジンのネットワークスタックから、スキャンのためにカーネル空間外で動作するプロセスにパケットを送信できます。したがって、Dr.Web Firewall for Linuxは、特殊な*Netlink*ソケットを介して、指定した番号のキュー*NFQUEUE*に接続し、処理で判定するために必要なパケットを入手します(Dr.Web Firewall for Linuxは、**NetFilter**に、*DROP: ドロップ*、*ACCEPT: 許可*、*REPEAT: 繰り返し*のいずれかの判定を伝える必要があります)。

パケットと接続のマーク

パケットをマークするため、Dr.Web Firewall for Linuxは、パケットおよび接続マークで使用可能な32ビットのうち次の3つを使用します。

- LDMビット(*Local Delivery Mark*)。マークにこのビットがあるパケットは、使用しているルーティングルールに基づいてローカルホストに送信されます。
- CPMビット(*Client Packets Mark*)。クライアント(接続開始側)とプロキシ(Dr.Web Firewall for Linuxなど)間の接続を示します。
- SPMビット(*Server Packets Mark*)。プロキシ(Dr.Web Firewall for Linuxなど)とサーバー(接続受信側)の間の接続を示します。

LDP、SDP、MCPビットは、ルーティング接続を実行する他のアプリケーションがパケットをマークするために使用していない任意の*さまざまな*ビットにすることができます。デフォルトでは、Dr.Web Firewall for Linuxは適切な(他のアプリケーションでは使用されていない)ビットを自動的に選択します。

ルートとルーティングポリシー(ip rule、ip route)

Dr.Web Firewall for Linuxを(あらゆる接続スキャンモードで)正しく動作させるには、100番のルーティングテーブルを使用する**ip** `rule`ルーティングポリシーをシステムに設定する必要があります。

```
from all fwmark <LDM>/<LDM> lookup 100
```

次のルートをテーブルに追加する必要があります。

```
local default dev lo scope host
```

このルーティングポリシーは、マークにLDMビットが入っているパケットが常にローカルホストに送信されることを保証します。

それ以降、XXXビットの *<XXX>*文字列は、2 ^{N}(*N*はパケットマーク内のXXXビットの序数)に等しい*16進*値になります。たとえば、最小(ゼロ)ビットがLDMビットとして選択されている場合は、*<LDM>* = 2^{0} = *0x1*になります。

NetFilter(iptables)のルール

Dr.Web Firewall for Linuxを(あらゆる接続スキャンモードで)正しく動作させるには、次の6つのルール(**iptables-save**出力コマンド形式で表示)が**NetFilter**システムコンポーネントの該当するチェーンのnatテーブルとmangleテーブルに存在している必要があります。

```
*nat

-A POSTROUTING -o lo -m comment --comment drweb-firewall -m mark --mark
<LDM>/<LDM> -j ACCEPT

*mangle

-A PREROUTING -m comment --comment drweb-firewall -m mark --mark
0x0/<CPM+SPM> -m connmark --mark <SPM>/<CPM+SPM> -j MARK --set-xmark
<LDM>/<LDM>
-A PREROUTING -p tcp -m comment --comment drweb-firewall -m mark ! --mark
<CPM+SPM>/<CPM+SPM> -m connmark --mark <CPM>/<CPM+SPM> -j TPROXY --on-port
<port> --on-ip <IP-address> --tproxy-mark <LDM>/<LDM>
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark
<CPM>/<CPM+SPM> -j CONNMARK --set-xmark <CPM>/0xffffffff
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark
<SPM>/<CPM+SPM> -j CONNMARK --set-xmark <SPM>/0xffffffff
-A OUTPUT -m comment --comment drweb-firewall -m mark --mark 0x0/<CPM+SPM>
-m connmark ! --mark 0x0/<CPM+SPM> -j MARK --set-xmark <LDM>/<LDM>
```

以下の説明では、0～5の番号がこれらのルールに割り当てられています(文書に記載されている順序で)。式 *<X+Y>*は、それぞれの数*X*と*Y*のビット数「OR」(合計)を意味します。

ルール番号2のパラメータ*<IP address>*と*<port>*は、Dr.Web Firewall for Linuxが監視された接続を管理するネットワークソケットを示します。

さらに、Dr.Web Firewall for Linux設定で監視接続モード(送信、受信、トランジット)を有効にする場合は、次の追加ルールが該当するチェーンのmangleテーブル(*OUTPUT*、*INPUT*、*FORWARD*)に存在する必要があります。

- *送信*(*OUTPUT*)接続を監視するには:

```
-A OUTPUT -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <ONum> --
queue-bypass
```

- *受信*(*INPUT*)接続を監視するには:

```
-A INPUT -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <INum> --
queue-bypass
```

- *トランジット(FORWARD)*接続を監視するには:

```
-A FORWARD -p tcp -m comment --comment drweb-firewall -m tcp --tcp-flags
SYN,ACK SYN -m mark --mark 0x0/<CPM+SPM> -j NFQUEUE --queue-num <FNum> --
queue-bypass
```

以下の説明では、6、7、8の番号がこれらのルールに割り当てられています(文書に記載されている順序で)。

ここで、*<ONum>*、*<INum>*、*<FNum>*は、Dr.Web Firewall for Linuxが対応する接続のインストールを示すパケット(SYNフラグが設定されているがACKフラグは含まれていないパケット)を待機している*NFQUEUE*内のキューの数です。

接続監視の順序

ルール6、7、8のいずれかに従って、対応する方向の新しいネットワーク接続を示すパケットは、CPMとSPMのどちらのビットでもマークされていない場合、**NetFilter**によって該当する*NFQUEUE*キューに入れられます。ここでパケットは、*nf_conntrack*インターフェースを介してDr.Web Firewall for Linuxによって読み取られます。ルール3と4は、接続を監視済みとしてマークします。つまり、接続マークに接続方向が設定されていることを示すビットがあります。このビット番号は、パケットマークのビット番号と一致します。その結果、ルール1、2、5に従って、この接続を介して送信されたパケットはDr.Web Firewall for Linuxによって配信されます。ルール0がnatテーブルの*POSTROUTING*チェーンの最上部に追加されるため、NATが設定されている場合、マークされたパケットのアドレスは送信されません(Dr.Web Firewall for Linuxの監視および接続処理ロジックに干渉するため)。

パケットがいずれかの*NFQUEUE*キューに現れると、Dr.Web Firewall for Linuxは、**NetFilter**で誤ったルールが設定されている場合に備えて、パケットの基本的な処理を実行します。次に、Dr.Web Firewall for Linuxは、ルール4に従い、それ自体の名前と、PSCとマークされたソケットを使用してサーバーへの接続を試みます。ローカル配信のルール5は適用されません。パケットはSPMでマークされており、このルールは*<CPM + SPM>*でマークされたパケットにのみ適用できるためです。

- *サーバーへの接続が失敗した場合*、Dr.Web Firewall for Linuxは、RSTビットを含むクライアントパケットを生成し、*<IP address>*:*<port>*のペアをリクエストされたサーバーのネットワークソケットのアドレスに置き換えます。DROP判定も*NFQUEUE*に送信されます。RSTビットを含むパケットの送信に使用されるソケットは*<CPM+SPM>*としてマークされているため、上記のルールはどれも適用されず、パケットは通常のルーティングルールに従ってクライアントに配信されます。
- *リモートサーバーへの接続が成功した場合*、Dr.Web Firewall for Linuxは、監視したSYNパケットをコピーし、*<LDM+CPM>*とマークされたソケットから再送信して、パケットをローカルネットワークソケットにリダイレクトします。LDMビットが設定されているため、指定したルーティングルールに従って出力インターフェースを選択すると、パケットは*loopback*インターフェースに追加されます。その後、**NetFilter** *PREROUTING*チェーンに追加され、そこでルール2が適用されます。したがって、パケットは変更されることなくネットワークソケットDr.Web Firewall for Linuxにリダイレクトされます。この機能により、Dr.Web Firewall for Linuxは接続アドレスの4つの要素すべて(パケット送信者のIPアドレスとポート、パケット受信者のIPアドレスとポート)を保存できます。

Dr.Web Firewall for Linuxがルール2に従って監視した接続を受信するネットワークソケットの場合、IP_TRANSPARENTオプションと*<LDM+CPM>*マークが設定され、このソケットからDr.Web Firewall for

Linuxによって送信されたパケットが*NFQUEUE*キューに分類されないようにします。クライアントが接続すると、保存されている4要素のアドレス(パケット送信者のIPアドレスとポート、パケット受信者のIPアドレスとポート)を使用して、ペアのソケットが検索されます。クライアントとサーバーの接続が確立されると、Luaの手順とDr.Web Firewall for Linux設定で指定したスキャンルールに従ってスキャンされます。スキャンが成功し、接続が安定している場合は、クライアント側とサーバー側を接続する関連ソケットのペアが転送データの分析のためにSpIDer Gateコンポーネントに転送されます。次のクライアントとサーバー間の対話は、メディエーターSpIDer Gateを介して確立されます。Dr.Web Firewall for Linuxは、クライアント側とサーバー側に関連付けられたソケットペアに加えて、確立された接続をスキャンするためのパラメータとルールをSpIDer Gateに送信します。

Dr.Web Firewall for Linux動作の概略図を以下に示します。

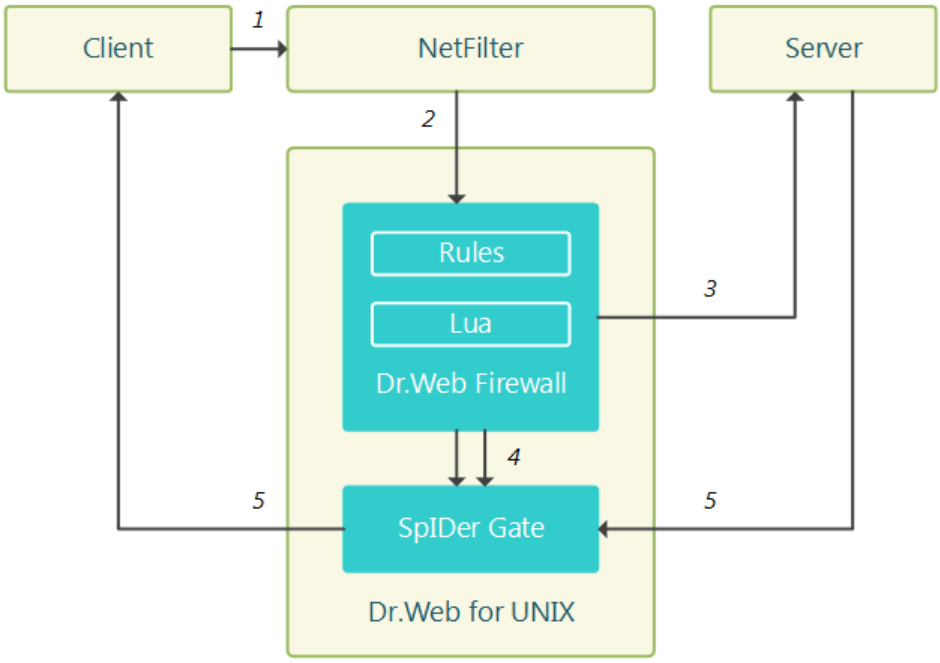

図12. コンポーネントの動作図

以下の接続処理のステップには番号が付いています。

1. クライアントがサーバーへの接続を試みます。
2. ルーティングルールに従って、**NetFilter**の接続をDr.Web Firewall for Linuxにリダイレクトします。
3. Dr.Web Firewall for Linuxは、クライアントの名前と接続スキャンを使用して、サーバーへの接続を試みます。
4. 接続のクライアント側とサーバー側、接続処理用のSpIDer Gate、およびスキャンの設定とルールに関連するソケットペアの送信。
5. メディエーターとしてSpIDer Gateを介したサーバーとクライアント間のデータ交換。

Dr.Web Firewall for Linuxを正しく動作させるためには、ルーティングテーブルに正しい数のビットマーク、*NFQUEUE*キュー、接続監視用のネットワークソケットアドレスを使用するこうしたルールが必要です。デフォルト設定では、コンポーネントは必要なルール設定を自動的に実行します。設定で接続の自動設定が無効になっている場合、コンポーネントを起動するときに必要なルールを手動で入力する必要があります。

コマンドライン引数

Dr.Web Firewall for Linuxを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-firewall [<options>]
```

Dr.Web Firewall for Linuxは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-firewall --help
```

このコマンドはDr.Web Firewall for Linuxに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-firewall`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[LinuxFirewall]`セクションで指定されている設定パラメータを使用します。

- コンポーネントパラメータ。
- トラフィックモニタリングとアクセスブロックのルール。

コンポーネントパラメータ

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-firewall • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-firewall • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-firewall
XtablesLockPath *{path to file}*	**iptables**（**NetFilter**）テーブルブロックファイルへのパス。パラメータ値が指定されていない場合は、/run/xtables.lockと/var/run/xtables.lockパスが検査されます。指定されたパスまたはデフォルトのパスにファイルが見つからない場合は、コンポーネントを起動したときにエラーが発生します。 **デフォルト値**：*（指定なし）*
InspectFtp *{On / Off}*	FTPプロトコルを介して転送されたデータをスキャンするかどうかを指示します。 *実際のデータスキャンは、指定されたスキャンルールに従って実行されます（下記参照）。* **デフォルト値**：On
InspectHttp *{On / Off}*	HTTPプロトコルを介して転送されたデータをスキャンするかどうかを指示します。 *実際のデータスキャンは、指定されたスキャンルールに従って実行されます（下記参照）。* **デフォルト値**：On
InspectSmtp *{On / Off}*	SMTPプロトコルを介して転送されたデータをスキャンするかどうかを指示します（インストールされている場合は、Dr.Web MailDコンポーネントが使用されます）。

	実際のデータスキャンは、指定されたスキャンルールに従って実行されます（下記参照）。 **デフォルト値：** `On`
InspectPop3 *{On / Off}*	POP3プロトコルを介して転送されたデータをスキャンするかどうかを指示します（インストールされている場合は、Dr.Web MailDコンポーネントが使用されます）。 *実際のデータスキャンは、指定されたスキャンルールに従って実行されます（下記参照）。* **デフォルト値：** `On`
InspectImap *{On / Off}*	IMAPプロトコルを介して転送されたデータをスキャンするかどうかを指示します（インストールされている場合は、Dr.Web MailDコンポーネントが使用されます）。 *実際のデータスキャンは、指定されたスキャンルールに従って実行されます（下記参照）。* **デフォルト値：** `On`
AutoconfigureIptables *{Yes / No}*	**iptables**インターフェースを介して**NetFilter**システムコンポーネントを設定するためのルール。 **使用可能な値：** • `Yes` - コンポーネントを起動するときに**NetFilter**のルールを設定し、動作を自動的に終了するときにルールを削除します（*推奨*）。 • `No` - ルールの自動設定を実行しません。必要なルールは、コンポーネントを起動する前に管理者が手動で追加し、動作が完了した後に削除する必要があります。 ⚠ **iptables**のルールの自動設定が許可されていない場合は、**iptables**に必要なルールがコンポーネント動作開始前に利用できるようになっている必要があります。 **デフォルト値：** `Yes`
AutoconfigureRouting *{Yes / No}*	**ip route**と**ip rule**のルーティングルールとポリシーの設定モード。 **使用可能な値：** • `Yes` - コンポーネントを起動するときに**ip route**と**ip rule**のルーティングルールとポリシーを設定し、動作を自動的に終了するときにルールを削除します（*推奨*）。 • `No` - ルールの自動設定を実行しません。必要なルールは、コンポーネントを起動する前に管理者が手動で追加し、動作が完了した後に削除する必要があります。

	 ルーティングルールとポリシーの自動設定が許可されていない場合は、**ip route**と**ip rule**に必要なルールがコンポーネント動作開始前に利用できるようになっている必要があります。 **デフォルト値**：`Yes`
`LocalDeliveryMark` *{integer / Auto}*	接続を監視するためにDr.Web Firewall for Linuxネットワークソケット（**`TproxyListenAddress`**パラメータで指定、下記参照）にリダイレクトされるパケットの *<LDM>*マーク。 **使用可能な値**： • *<integer>* - パケットの *<LDM>*マーク。2^Nに等しく、*N*がパケット内の`LDM`ビット数の場合、$0 \leq N \leq 31$。 • `Auto` - Dr.Web Firewall for Linuxはパケットマークの適切なビット数を自動的に選択できます（*推奨*）。  *<LDM>*番号を手動で割り当てるときは、（**NetFilter**経由を含む）ルート接続とパケットを管理する他のアプリケーションが、パケットマークで相応するビット数を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **`AutoconfigureIptables`** = `No`、**`AutoconfigureRouting`** = `No`の場合、指定された *<LDM>*番号を手動で追加する必要があるルーティングルールで使用してください。 **デフォルト値**：`Auto`
`ClientPacketsMark` *{integer / Auto}*	接続を開始するクライアントとDr.Web Firewall for Linuxの間で転送されるパケットの *<CPM>*マーク。 **使用可能な値**： • *<integer>* - パケットの *<CPM>*マーク。2^Nに等しく、*N*がパケット内の`CPM`ビット数の場合、$0 \leq N \leq 31$。 • `Auto` - Dr.Web Firewall for Linuxはパケットマークの適切なビット数を自動的に選択できます（*推奨*）。

	 *<CPM>*番号を手動で割り当てるときは、(**NetFilter**経由を含む)ルート接続とパケットを管理する他のアプリケーションが、パケットマークで、対応するビット数を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = Noの場合、指定された*<CPM>*番号を手動で追加する必要があるルーティングルールで使用する必要があります。 **デフォルト値**: Auto
ServerPacketsMark *{integer | Auto}*	*Dr.Web Firewall for Linuxと接続を受信するサーバーの間で転送されるパケットの*<SPM>マーク。 **使用可能な値**: • *<integer>* - パケットの *<SPM>*マーク。2^Nに等しく、*N*がパケット内のSPMビット数の場合、$0 \leq N \leq 31$。 • Auto - Dr.Web Firewall for Linuxはパケットマークの適切なビット数を自動的に選択できます(*推奨*)。  *<SPM>*番号を手動で割り当てるときは、(**NetFilter**経由を含む)ルート接続とパケットを管理する他のアプリケーションが、パケットマークで相応するビット数を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = No、**AutoconfigureRouting** = Noの場合、指定された*<SPM>*番号は手動で追加する必要があるルーティングルールで使用してください。 **デフォルト値**: Auto
TproxyListenAddress *{network socket}*	Dr.Web Firewall for Linuxが監視した接続を受信するネットワークソケット(*<IP address>*:*<port>*)。ポート0を指定すると、システムによって自動的に選択されます。

	 該当するソケットが他のアプリケーションによって使用されていないことを確認する必要があります。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = Noの場合、指定されたIPアドレスとポートは手動で追加する必要があるルーティングルールで使用してください。 **デフォルト値**：127.0.0.1:0
OutputDivertEnable *{Yes / No}*	受信接続モード（つまりローカルホストに接続があるリモートのアプリケーションによって開始された接続）の監視。 **使用可能な値**： • Yes - 送信接続の監視と処理を行います。 • No - 送信接続の監視と処理は行いません。  この設定は、**AutoconfigureIptables** = Noの場合に手動で追加または削除されるルーティングルール番号5を追加または削除します。 **デフォルト値**：No
OutputDivertNfqueueNumber *{integer / Auto}*	Dr.Web Firewall for Linuxが送信接続を開始するSYNパッケージを取得するキュー番号 *NFQUEUE*。 **使用可能な値**： • <integer> - *NFQUEUE*で監視された送信接続のSYNパケットを監視するためのキュー番号 *<ONum>*。 • Auto - Dr.Web Firewall for Linuxは適切なキュー番号を自動的に選択できます（*推奨*）。  *<ONum>*番号を手動で割り当てるときは、（**NetFilter**ルール経由を含む）接続とパケットを管理する他のアプリケーションが、相応するキュー番号を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = Noの場合、指定された *<ONum>*番号を手動で追加する必要があるルーティングルール番号5で使用する必要があります。

	デフォルト値：Auto
OutputDivertConnectTransparently *{Yes / No}*	送信接続のために監視されたパケットの送信者（クライアント）のIPアドレスを使用して受信者（サーバー）に接続するためのエミュレーションモード。 **使用可能な値**： • Yes - 接続を監視するときに、自分の代わりに接続をリクエストしたクライアントのアドレスを使用してサーバーに接続します • No - Dr.Web Firewall for Linuxアドレスからサーバーに接続します。 *クライアントとDr.Web Firewall for Linuxアドレスは通常、送信接続監視モードでは同じであるため、デフォルト値はNoになります。* **デフォルト値**：No
InputDivertEnable *{Yes / No}*	受信接続モード（つまりローカルホストに接続があるリモートホストのアプリケーションによって開始された接続）の監視。 **使用可能な値**： • Yes - 受信接続の監視と処理を行います。 • No - 受信接続の監視と処理を行いません。 ⚠ この設定は、**AutoconfigureIptables** = Noの場合に手動で追加または削除されるルーティングルール番号6を追加または削除します。無効な値を指定した場合、コンポーネントの起動は失敗します。 **デフォルト値**：No
InputDivertNfqueueNumber *{integer / Auto}*	Dr.Web Firewall for Linuxが受信接続を開始するSYNパケットを取得するキュー番号 *NFQUEUE*。 **使用可能な値**： • <integer> - *NFQUEUE*で監視された送信接続のSYNパケットを監視するためのキュー番号 *<INum>*。 • Auto - Dr.Web Firewall for Linuxは適切なキュー番号を自動的に選択できます（*推奨*）。

	 *<INum>*番号を手動で割り当てるときは、(**NetFilter**ルール経由を含む)接続とパックを管理する他のアプリケーションが、相応するキュー番号を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = Noの場合、指定された*<INum>*番号を手動で追加する必要があるルーティングルール番号6で使用する必要があります。 **デフォルト値**:Auto
InputDivertConnectTransparently *{Yes / No}*	受信接続のために監視されたパケットの送信者(クライアント)のIPアドレスを使用して受信者(サーバー)に接続するためのエミュレーションモード。 **使用可能な値**: • Yes - 接続を監視するときに、自分の代わりに接続をリクエストしたクライアントのアドレスを使用してサーバーに接続します • No - Dr.Web Firewall for Linuxアドレスからサーバーに接続します。 *受信接続監視モードでは、すべてのトラフィックがDr.Web Firewall for Linuxを通過するため、不正なクライアントのアドレスを使用してサーバーに安全に接続する可能性があります。これがデフォルト値がYesである理由です。* **デフォルト値**:Yes
ForwardDivertEnable *{Yes / No}*	トランジット接続モード(つまりその他のリモートホストに接続があるリモートホストのアプリケーションによって開始された接続)の監視。 **使用可能な値**: • Yes - トランジット接続の監視と処理を行います。 • No - トランジット接続の監視と処理は行いません。  この設定は、**AutoconfigureIptables** = Noの場合に手動で追加または削除されるルーティングルール番号7を追加または削除します。 **デフォルト値**:No
ForwardDivertNfqueueNumber	Dr.Web Firewall for Linuxがトランジット接続を開始するSYNパケットを取得するキュー番号*NFQUEUE*。

{integer | Auto}	**使用可能な値**: • `<integer>` - *NFQUEUE*で監視されたトランジット接続のSYNパケットを監視するためのキュー番号 *<FNum>*。 • `Auto` - Dr.Web Firewall for Linuxは適切なキュー番号を自動的に選択できます(*推奨*)。 ⚠ *<FNum>*番号を手動で割り当てるときは、(**NetFilter**ルール経由を含む)接続とパックを管理する他のアプリケーションが、相応するキュー番号を使用していないことを確認してください。無効な値を指定した場合、コンポーネントの起動は失敗します。 **AutoconfigureIptables** = `No`の場合、指定された *<FNum>*番号を手動で追加する必要があるルーティングルール番号7で使用してください。 **デフォルト値**: `Auto`
ForwardDivertConnectTransparently *{Yes | No}*	トランジット接続のために監視されたパケットの送信者(クライアント)のIPアドレスを使用して受信者(サーバー)に接続するためのエミュレーションモード。 **使用可能な値**: • `Yes` - 接続を監視するときに、自分の代わりに接続をリクエストしたクライアントのアドレスを使用してサーバーに接続します • `No` - Dr.Web Firewall for Linuxアドレスからサーバーに接続します。 *トランジット接続監視モードでは、すべてのトラフィックがDr.Web Firewall for Linuxがインストールされているのと同じホスト(ルーター)を通過するという保証はないため、正しい動作のデフォルト値は*`No`*になります。保護されたアプリケーションが同じルーターを使用することがネットワーク設定によって保証されている場合、このパラメータを*`Yes`*に設定できます。この場合は、Dr.Web Firewall for Linuxがサーバーに接続するときは常にクライアントのアドレスへの接続を評価します。* **デフォルト値**: `No`
ExcludedProc *{path to file}*	*プロセスのホワイトリストとして使用することができる*プロセスのリスト、つまりネットワーク活動を監視してはならないプロセスのリスト。 *リストをパラメータ値として指定できます。リストの値は、コンマ(引用符内の各値)で区切る必要があります。パラメータはセクションで複数回指定できます(この場合、そのすべての値が1つのリストにまとめられます)。*

	例：プロセスのリストに**wget**と**curl**を追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[LinuxFirewall]` `ExcludedProc = "/usr/bin/wget", "/usr/bin/curl"` • 2つの文字列（文字列ごとに1つの値） `[LinuxFirewall]` `ExcludedProc = /usr/bin/wget` `ExcludedProc = /usr/bin/curl` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset LinuxFirewall.ExcludedProc -a /usr/bin/wget` `# drweb-ctl cfset LinuxFirewall.ExcludedProc -a /usr/bin/curl` ⚠ このパラメータで示されるプロセスリストの実際の使用方法は、Dr.Web Firewall for Linuxに定義されているスキャンルールでの使用*方法*によって決まります。 デフォルトルールのリスト（以下参照）は、リストからの全プロセスのトラフィックが*スキャンせずに*許可されることを保証します。 **デフォルト値**：*（未設定）*
UnwrapSsl *{Boolean}*	SSL/TLS接続を介して転送された暗号化トラフィックをスキャンするように指示します。

	 最近の実行例では、この変数の値は保護されたトラフィックの処理に影響しません。処理を管理するには、SET Unwrap_SSL = true/falseアクションを含むルールを作成する必要があります（以下参照）。 **drweb-ctl**ユーティリティのcfsetコマンドまたはWebインターフェースを使用してこのパラメータの値を変更した場合、影響を受ける依存ルールが自動的に適用されます。 **デフォルト値**：No
BlockInfectionSource *{Boolean}*	悪意のあるソフトウェア（*InfectionSource*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：Yes
BlockNotRecommended *{Boolean}*	非推奨サイト（*NotRecommended*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：Yes
BlockAdultContent *{Boolean}*	アダルトコンテンツ（*AdultContent*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No

BlockViolence *{Boolean}*	暴力的描写（*Violence*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No
BlockWeapons *{Boolean}*	武器に関するWebサイト（*Weapons*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No
BlockGambling *{Boolean}*	ギャンブルのWebサイト（*Gambling*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No
BlockDrugs *{Boolean}*	麻薬に関するWebサイト（*Drugs*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No
BlockObsceneLanguage *{Boolean}*	卑猥な表現（*ObsceneLanguage*カテゴリーに含まれる）を含むWebサイトへの接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。

	`` url_category in "LinuxFirewall.BlockCategory" : Block as _match ``` **デフォルト値**：No
BlockChats *{Boolean}*	チャットWebサイト（*Chats*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "LinuxFirewall.BlockCategory" : Block as _match ``` **デフォルト値**：No
BlockTerrorism *{Boolean}*	テロリズムに関するWebサイト（*Terrorism*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "LinuxFirewall.BlockCategory" : Block as _match ``` **デフォルト値**：No
BlockFreeEmail *{Boolean}*	無料メールサービスのWebサイト（*FreeEmail*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "LinuxFirewall.BlockCategory" : Block as _match ``` **デフォルト値**：No
BlockSocialNetworks *{Boolean}*	ソーシャルネットワーキングサイト（*SocialNetworks*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 ``` url_category in "LinuxFirewall.BlockCategory" : Block as _match ``` **デフォルト値**：No

BlockDueToCopyrightNotice *{Boolean}*	著作権者のリクエストに従って追加されたWebサイト（*DueToCopyrightNotice*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：`Yes`
BlockOnlineGames *{Boolean}*	オンラインゲームWebサイト（*OnlineGames*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：`No`
BlockAnonymizers *{Boolean}*	アノニマイザーWebサイト（*Anonymizers*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：`No`
BlockCryptocurrencyMiningPools *{Boolean}*	仮想通貨マイニングのための一般的なサービスへのアクセスを提供するWebサイト（*CryptocurrencyMiningPool*カテゴリーに含まれる）への接続試行をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：`No`
BlockJobs *{Boolean}*	求人検索Webサイト（*求人*カテゴリーに含まれます）への接続試行をブロックするように指示します。

	ブロックをアクティブにするには、次のルールを設定に追加する必要があります（下記の詳細を参照）。 `url_category in "LinuxFirewall.BlockCategory" : Block as _match` **デフォルト値**：No
Whitelist *{domain list}*	*ホワイトリストとして使用できる*ドメインのリスト（つまり、これらのドメインがブロックされたカテゴリーに含まれている場合でも、ユーザーの接続が許可されているドメインのリスト。さらに、このリストに示されているドメインのすべてのサブドメインへユーザーがアクセスすることが許可されます）。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ドメイン`example.com`と`example.net`のリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[LinuxFirewall]` `Whitelist = "example.com", "example.net"` • 2つの文字列（文字列ごとに1つの値） `[LinuxFirewall]` `Whitelist = example.com` `Whitelist = example.net` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset LinuxFirewall.Whitelist -a example.com` `# drweb-ctl cfset LinuxFirewall.Whitelist -a example.net`

	 このパラメータで示されるドメインリストの実際の使用方法は、Dr.Web Firewall for Linuxに定義されているスキャンルールでの使用*方法*によって決まります。 デフォルトルールのリスト（下記参照）は、ブロックされるWebソースカテゴリーのリストのドメインがこのリストに含まれている場合でも、このリストのドメイン（およびそのサブドメイン）へのアクセスが提供されることを保証します（ただしHTTPプロトコルを経由したサーバーへのリクエストの場合のみ）。さらに、このデフォルトのルールセットは、ホワイトリストドメインからダウンロードしたデータが*脅威に対してスキャンされる*ことを保証します（データはレスポンスで返され、変数`direction`には値`response`があるため）。 **デフォルト値**：*（未設定）*
`Blacklist` *{domain list}*	*ブラックリストとして使用できる*ドメインのリスト（つまり、これらのドメインがブロックされたカテゴリーに含まれていない場合でも、ユーザーの接続が禁止されているドメインのリスト。さらに、このリストに示されているドメインのすべてのサブドメインへユーザーがアクセスすることが禁止されます）。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ドメイン`example.com`と`example.net`のリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[LinuxFirewall]` `Blacklist = "example.com", "example.net"` • 2つの文字列（文字列ごとに1つの値） `[LinuxFirewall]` `Blacklist = example.com` `Blacklist = example.net` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。

	`` # drweb-ctl cfset LinuxFirewall.Blacklist -a example.com # drweb-ctl cfset LinuxFirewall.Blacklist -a example.net ``` このパラメータで示されるドメインリストの実際の使用方法は、Dr.Web Firewall for Linuxに定義されているスキャンルールでの使用*方法*によって決まります。 デフォルトルールのリスト（以下参照）は、このリストのドメイン（およびそのサブドメイン）へのアクセスがHTTPプロトコル上で常に禁止されることを保証します。このドメインがホワイトリストとブラックリストに同時に追加される場合、デフォルトのルールにより、そのドメインへのユーザーアクセスは確実にブロックされます。 **デフォルト値**：*（未設定）*
ScanTimeout *{time interval}*	SpIDer Gateによって開始された1つのファイルに対するスキャンのタイムアウト。 *1秒から1時間の範囲の値を指定できます。* **デフォルト値**：`30s`
HeuristicAnalysis *{On / Off}*	SpIDer Gateによって開始されたファイルのスキャン中に脅威を検出するためにヒューリスティック解析を使用するかどうかを指定します。ヒューリスティック解析における検出の信頼性は高いのですが、ウイルススキャンに時間がかかります。 *ヒューリスティックアナライザによって検出された脅威に適用されるアクションは、**BlockSuspicious**パラメータ値として指定します。* **使用可能な値**： • `On` - スキャン時にヒューリスティック解析を使用するように指示します。 • `Off` - ヒューリスティック解析を使用しないように指示します。 **デフォルト値**：`On`
PackerMaxLevel *{integer}*	圧縮されたオブジェクトスキャン時の最大ネスティングレベル。SpIDer Gateによるファイルスキャン時に、下位のネストレベルにあるすべてのオブジェクトがスキップされます。

	0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。 **デフォルト値**：8
ArchiveMaxLevel *{integer}*	アーカイブスキャン時の最大ネスティングレベル。SpIDer Gateによるファイルスキャン時に、下位のネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**：8
MailMaxLevel *{integer}*	メールメッセージとメールボックスをスキャンするときの最大ネストレベル。SpIDer Gateによるファイルスキャン時に、下位のネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**：8
ContainerMaxLevel *{integer}*	コンテナ（HTMLページなど）をスキャンするときの最大ネストレベル。SpIDer Gateによるファイルスキャン時に、下位のネストレベルにあるすべてのオブジェクトがスキップされます。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* **デフォルト値**：8
MaxCompressionRatio *{integer}*	圧縮/パックされたオブジェクトの最大圧縮率（非圧縮サイズと圧縮サイズの比率）。オブジェクトの比率が制限を超えると、そのオブジェクトはSpIDer Gateによって開始されたファイルスキャン中にスキップされます。 *圧縮率には2よりも小さい値は指定できません。* **デフォルト値**：500
BlockKnownVirus *{Boolean}*	データに既知の脅威が含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：Yes
BlockSuspicious *{Boolean}*	データに未知の脅威（ヒューリスティックアナライザによって検出されたもの）が含まれる場合、そのデータの受信または送信をブロックするように指示します。

	ブロックをアクティブにするには、次のルールを設定に追加する必要があります（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：`Yes`
BlockAdware *{Boolean}*	データにアドウェアが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：`Yes`
BlockDialers *{Boolean}*	データにダイアラープログラムが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：`Yes`
BlockJokes *{Boolean}*	データにジョークプログラムが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：`No`
BlockRiskware *{Boolean}*	データにリスクウェアが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match`

	デフォルト値：No
BlockHacktools *{Boolean}*	データにハッキングツールが含まれる場合、そのデータの受信または送信をブロックするように指示します。 *ブロックをアクティブにするには、次のルールを設定に追加する必要があります*（下記の詳細を参照）。 `threat_category in "LinuxFirewall.BlockThreat" : Block as _match` **デフォルト値**：No
BlockUnchecked *{Boolean}*	データがスキャンできない場合、そのデータの受信または送信をブロックするように指示します。 このパラメータ値は、エラーのためにtrueまたはfalseに評価することが不可能なルールの処理に影響します。Noを指定した場合、ルールは実行されていないルールとしてスキップされます。Yesを指定した場合、Block as BlackListアクションが実行されます。 **デフォルト値**：No
InterceptHook *{path to file / Lua function}*	Luaで接続を処理するためのスクリプトまたはそのスクリプトを含むファイルへのパス（Luaでの接続処理セクションを参照）。 *使用できないファイルを指定すると、コンポーネントを読み込む際にエラーが表示されます。* **デフォルト値**： `local dwl = require 'drweb.lookup'` `function intercept_hook(ctx)` `  -- do not check if group == Root.TrustedGroup` `  if ctx.divert == "output" and ctx.group == "drweb"` `  then` `  return "pass"` `  end` `  -- do not check connections from privileged ports` `  -- except FTP active mode` `  if ctx.src.port >= 0 and ctx.src.port <= 1024` `  and ctx.src.port ~= 20` `  then` `  return "pass"`

```
  end

  return "check"
end
```

接続スキャンの設定を変更しても、変更を加える前にアプリケーションによってすでに確立されている接続のスキャンには影響しません。それらをすでに実行中のアプリケーションに適用する必要がある場合は、アプリケーションを再起動するなどして、アプリケーションを強制的に切断してから再度接続する必要があります。

トラフィックモニタリングとアクセスブロックのルール

このセクションには、上のパラメータの他、11個の*ルールセット*`RuleSet*`（`RuleSet0`、...、`RuleSet10`）も含まれます。これらは、トラフィックスキャン、Webリソースへのユーザーアクセスのブロック、インターネットからのコンテンツのダウンロードのブロックを直接管理します。条件の一部の値（IPアドレス範囲、Webサイトカテゴリーのリスト、Webソースのブラックリストとホワイトリストなど）については、テキストファイルからロードされる値の置き換えがあり、LDAPを介して外部データソースから抽出されます（Dr.Web LookupDコンポーネントを使用）。接続を設定する際には、最終的な解決を含むルールが見つかるまで、すべてのルールが昇順で検査されます。ルールリストのギャップは無視されます。

ルールについては、トラフィックモニタリングのルールの付録D「トラフィックモニタリングのルール」セクションで詳しく説明されています。

ルールを表示して編集する

ルールリストを簡単に編集するためにブランクのものが残されています。つまり、ルールが指定されていない`RuleSet`*<i>*があります（*<i>* - `RuleSet`ルールセット番号）。`RuleSet`*<i>*以外の項目を追加することは*できません*`RuleSet`*<i>*の要素内のルールは追加および削除できます。ルールの表示と編集は、次のいずれかの方法で実行できます。

- （任意のテキストエディターで）設定ファイル設定ファイルを表示、編集する（このファイルにはデフォルトと異なるパラメータのみが保存されます）。
- Web管理インターフェース経由（インストールされている場合）。
- コマンドラインベースのインターフェースを介して - Dr.Web Ctl（**drweb-ctl** `cfshow`および**drweb-ctl** `cfset` コマンド）。

ルールを編集して設定ファイルを変更した場合は、これらの変更を適用するためにDr.Web for UNIX Internet Gatewaysを再起動します。それには、**drweb-ctl**`reload`コマンドを使用します。

コマンド**drweb-ctl** `cfshow`を使用してルールを表示します。

ルールセット`LinuxFirewall.RuleSet1`のコンテンツを表示するには、このコマンドを使用します。

```
# drweb-ctl cfshow LinuxFirewall.RuleSet1
```

drweb-ctl cfsetコマンドを使用してルールを編集します（以降、*<rule>* - ルールのテキスト）。

- **LinuxFirewall.RuleSet1**セットのすべてのルールを新しいルールで置き換えます。

```
# drweb-ctl cfset LinuxFirewall.RuleSet1 '<rule>'
```

- **LinuxFirewall.RuleSet1**ルールセットに新しいルールを次のように追加します。

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 '<rule>'
```

- **LinuxFirewall.RuleSet1**セットから特定のルールを次のように削除します。

```
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 '<rule>'
```

- **LinuxFirewall.RuleSet1**ルールセットを次のようにデフォルトの状態にリセットします。

```
# drweb-ctl cfset -r LinuxFirewall.RuleSet1
```

drweb-ctlツールを使用してルールのリストを編集するときは、追加するルールのテキストを一重引用符または二重引用符で囲み、ルールのテキスト内にある二重引用符の前にはバックスラッシュ（「\」）をエスケープ文字として使用します（ルール自体のテキストに二重引用符が含まれている場合）。

設定の**RuleSet***<i>*変数には、ルールの格納について次のような特徴があります。

- 無条件ルールを追加するときは、条件部分とコロンを省略できます。ただし、そのようなルールは常にルールのリストに文字列「 : *<action>*」として格納されます。
- 複数のアクションを含むルール（「*<condition>* : *<action 1>*, *<action 2>*」など）を追加すると、そのようなルールは基本ルールのチェーン「*<condition>* : *<action 1>*」と「*<condition>* : *<action 2>*」に変更されます。
- ロギングまたはルールは、条件部分で条件の離接（論理和）を許可しないため、論理和を実装するには、各ルールに離接語条件がある条件で一連のルールを記録する必要があります。

接続をスキップするための無条件ルール（Passアクション）を**LinuxFirewallRuleSet1**セットに追加するには、次のコマンドのみを実行します。

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'Pass'
```

ただし、指定したルールセットからこのルールを削除するには、次のコマンドを実行する必要があります。

```
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 ' : Pass'
```

LinuxFirewall.RuleSet1ルールを、接続用の標準テンプレートへのパスを未解決アドレスから変更してブロックを実行するルールセットに追加するには、次のコマンドを実行する必要があります。

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : set http_template_dir = "mytemplates", Block'
```

ただし、このコマンドでは指定したセットに*2つのルール*が追加されるため、ルールのセットからこれらのルールを削除するには、次の2つのコマンドを実行する必要があります。

```
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : set http_template_dir = "mytemplates"'
# drweb-ctl cfset -e LinuxFirewall.RuleSet1 'src_ip not in
file("/etc/trusted_ip") : Block'
```

「悪意のあるオブジェクト*KnownVirus*や*テロリズム*カテゴリーからのURLが検出された場合はブロックする」などのルールを**LinuxFirewall.RuleSet1**ルールセットに追加するには、このルールセットに次の2つのルールを追加する必要があります。

```
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'threat_category in (KnownVirus)
: Block as _match'
# drweb-ctl cfset -a LinuxFirewall.RuleSet1 'url_category in (Terrorism) :
Block as _match'
```

ルールのセットからこれらのルールを削除する場合にも、2つのコマンドを実行する必要があります（上の例を参照）。

デフォルトのルールセット

デフォルトでは、次のブロックルールセットが指定されています。

```
RuleSet0 =
RuleSet1 = divert output : set HttpTemplatesDir = "output"
RuleSet1 = divert output : set MailTemplatesDir = "firewall"
RuleSet1 = divert input : set HttpTemplatesDir = "input"
RuleSet1 = divert input : set MailTemplatesDir = "server"
RuleSet1 = proc in "LinuxFirewall.ExcludedProc" : Pass
RuleSet1 =  : set Unwrap_SSL = false
RuleSet2 =
RuleSet3 =
RuleSet4 =
RuleSet5 = protocol in (Http), direction request, url_host in
"LinuxFirewall.Blacklist" : Block as BlackList
RuleSet5 = protocol in (Http), direction request, url_host in
"LinuxFirewall.Whitelist" : Pass
RuleSet6 =
RuleSet7 = protocol in (Http), direction request, url_category in
"LinuxFirewall.BlockCategory" : Block as _match
RuleSet8 =
RuleSet9 = protocol in (Http), divert input, direction request,
threat_category in "LinuxFirewall.BlockThreat" : Block as _match
RuleSet9 = protocol in (Http), direction response, threat_category in
"LinuxFirewall.BlockThreat" : Block as _match
RuleSet9 = protocol in (Smtp), threat_category in
"LinuxFirewall.BlockThreat" :REJECT
RuleSet9 = protocol in (Smtp), url_category in "LinuxFirewall.BlockCategory"
: REJECT
RuleSet9 = protocol in (Smtp), total_spam_score gt 0.80 : REJECT
RuleSet9 = protocol in (Pop3, Imap), threat_category in
"LinuxFirewall.BlockThreat" :REPACK as _match
RuleSet9 = protocol in (Pop3, Imap), url_category in
```

```
"LinuxFirewall.BlockCategory" :REPACK as _match
RuleSet9 = protocol in (Pop3, Imap), total_spam_score gt 0.80 :REPACK as
_match
RuleSet10 =
```

最初のルールは、**ExcludedProc**パラメータ（上を参照）で指定されたプロセスによって接続が確立された場合、他の条件を検査せずに接続をスキップすることを示します。次のルール（条件なしで実行されます）は保護された接続のラップ解除をブロックします。このルールとその下位にあるすべてのルールは、接続が除外されたプロセスで設定されていない場合にのみ考慮されます。また、後続のすべてのルールはプロトコルに依存するため、保護された接続のラップ解除が無効になっている場合は、条件がtrueと評価されるかどうかを定義できないため、ルールは実行されません。

次のルールは、デフォルトのルールセットにおけるHTTP接続に対する処理の内容です。

1. 接続が確立されたホストがブラックリストに含まれている場合、そのホストがブラックリストに含まれているため、接続はブロックされます。他のスキャンは実行されません。
2. ホストがホワイトリストに含まれている場合、接続はスキップされ、他のスキャンは実行されません。
3. クライアントからリクエストされたURLが、アクセスが望ましくないとマークされているWebリソースのカテゴリーに含まれている場合は、脅威が検出されたために接続がブロックされます。他のスキャンは実行されません。
4. HTTP経由の脅威があるリモートホストから受信したレスポンスに、ブロックされたカテゴリーに属する脅威が含まれる場合、その脅威が検出されたために接続がブロックされます。他のスキャンは実行されません。
5. ローカルホストからリモートホストに移行したデータに、ブロックされたカテゴリーに属する脅威が含まれる場合、その脅威が検出されたために接続がブロックされます。他のスキャンは実行されません。

この5つのルールは、**InspectHttp**パラメータで**オン**が指定されている場合にのみ機能します。それ以外の場合、これらのルールはどれも機能しません。

RuleSet9で指定される次の6つのルールは、メールプロトコルを介して送受信されるデータのスキャンを管理します。これらのルールは、送信されたメール（SMTP、POP3、またはIMAPプロトコル経由）に、スパムとしてブロックまたは分類する必要があるカテゴリーに属する添付ファイルまたはURLが含まれていることが検出された場合に有効になります（Dr.Web ASEパーセンテージは0.8以上）。メールがSMTPプロトコルを介して送信される場合、メールの伝送（つまり送信または受信）はブロックされますが、IMAPとPOP3プロトコルの場合は、メールはそのコンテンツから悪意のあるコンテンツを除去するために処理（「再パッケージング」）されます。

メールのスパムの兆候をスキャンするためのコンポーネントであるDr.Web Anti-Spamが使用できない場合、スパムの兆候をスキャンするためのメールスキャンは実行されません。この場合、スパムレベルのスキャン（値`total_spam_score`）を含むルールは使用できません。

メール処理ルールは、相応する**Inspect**<*EmailProtocol*>パラメータに`On`が指定されている場合にのみ実行されることに注意してください。それ以外の場合、これらのルールはどれも実行されません。さらに、送信されたメールのマルウェアの添付ファイルを調べるためにメールスキャン用のDr.Web MailDコンポーネントをインストールする必要があります。コンポーネントがインストールされていない場合、*「検査できません」*というエラーのため、送信されたメールはブロックされます。検査できないメッセージの送信を許可するには、**BlockUnchecked** = `No`パラメータを設定します（上記を参照）。また、メールスキャンコンポーネントがインストールされていない場合は、**InspectSmtp**、**InspectPop3**、**InspectImap**パラメータに`No`を指定することをお勧めします。

トラフィックモニタリングとアクセスブロックのルールの例

1. IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーに*Chats*以外のすべてのカテゴリーのWebサイトへのHTTPアクセスを許可する。

```
protocol in (HTTP), src_ip in (10.10.0.0/24), url_category not in (Chats)
: Pass
```

ルール

```
protocol in (HTTP), url_host in "LinuxFirewall.Blacklist" : Block as
BlackList
```

が、指定されたルールより上のルールリストに割り当てられている場合、IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーは、ブラックリストのドメイン、つまりパラメータ`LinuxFirewall.Blacklist`にリストされているドメインへのアクセスもブロックされます。また、このルールが下に割り当てられている場合、IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーは、ブラックリストのWebサイトにもアクセスできます。

解決`Pass`は最終的なものであるため、これ以上のルールはチェックされず、したがってダウンロードされたデータのウイルススキャンも実行されません。IPアドレス範囲が*10.10.0.0*～*10.10.0.254*のユーザーに、ブラックリストに含まれていない*Chats*以外のすべてのカテゴリーのWebサイトへのアクセスを許可し、同時に脅威のダウンロードをブロックするには、次のルールを使用します。

```
protocol in (HTTP), url_category not in (Chats), url_host not in
"LinuxFirewall.Blacklist", threat_category not in
"LinuxFirewall.BlockCategory" : Pass
```

2. *インターネットからダウンロードした*ビデオファイルのコンテンツ（つまり、MIMEタイプが「`video/*`」のデータ（`*`はMIMEクラス`video`の任意のタイプ））のスキャンを実行しない。

```
direction response, content_type in ("video/*") : Pass
```

ローカルコンピューターから読み込んだファイル（MIMEタイプが「`video/*`」のファイルを含む）は、*レスポンス*ではなく*リクエスト*で送信されるため（つまり変数`direction`には値`request`があるため）にスキャンされます。

Luaでの接続処理

このセクションの内容：

- 概要
- 接続処理のスクリプトにおける要件
- 例
- 使用中のテーブル
- 利用可能な補助モジュール

概要

Dr.Web Firewall for LinuxはLuaのプログラムインタプリタを介して対話をサポートします（バージョン5.3.4が使用され、Dr.Web for UNIX Internet Gatewaysと一緒に提供されます）。Luaで書かれたスクリプトは、解析のためにSpIDer Gateに送信される前に、事前接続スキャンのためにコンポーネントによって使用されます。

Dr.Web Firewall for Linux設定（**`InterceptHook`**パラメータ内）が次のパスで指定されている場合、Luaスクリプトを使用した接続解析が、接続スキャンのスクリプトを含む、Luaで書かれたファイルに実行されます。それ

以外の場合は、コンポーネント設定で指定されているデフォルト設定と処理ルール（**RuleSet***パラメータ）を使用して接続処理が実行されます。

接続処理用のLuaスクリプトのその他の例については、次のリンクからご確認ください。
https://github.com/DoctorWebLtd/drweb-lua-examples/tree/master/firewall。

接続処理のスクリプトにおける要件

このファイルには、接続スキャンモジュールのエントリポイントであるグローバル関数が含まれている必要があります（Dr.Web Firewall for Linuxは、新しく受信した接続を処理するためにこの関数を呼び出します）。処理関数は、次の呼び出し規則を満たす必要があります。

1. *関数名* - `intercept_hook`;
2. *引数*は、Luaコンテキストテーブルのみです（処理された接続の関数からの情報へのアクセスを提供します。以下の表の説明を参照してください）。
3. *戻り値*は、以下の表の文字列値のみです。

値	判定の説明
`pass`	SpIDer Gateによるスキャンを行わずに接続をスキップする
`check`	SpIDer Gateを使用して接続をスキャンする
`drop`	パケット損失で接続をブロックする
`reject`	接続を破棄する（接続を開始し、*RST*フラグ付きのTCPパケットを受信するクライアント）

例

1. すべての接続について無条件に`pass`判定（スキップ）を返すLuaスクリプト。

```
-- Function of connection scanning written by the user
function intercept_hook(ctx)
  return "pass" -- do not scan the connection
end
```

2. Dr.Web Firewall for Linuxに対し、`drweb`グループのユーザー権限で実行されているアプリケーションからの送信ローカル接続、または（接続の所有者とその方向に関係なく）権限のあるポートから開始された接続、またはローカルネットワークからのIPアドレスから始まる接続を除く、確立された接続すべてをスキャンのために送信するように指示するスクリプト。

```
function intercept_hook(ctx)
  -- Do not scan connections, initiated from the local
  -- host (divert == "output") by application under the name of
  -- "drweb" (group == "[drweb]")
  if ctx.divert == "output" and ctx.group == "drweb" then
    return "pass"
  end

  -- Do not scan connections, initiated from
  -- privileged ports (range is from 0 to 1024)
  if ctx.src.port >= 0 and ctx.src.port <= 1024 then
    return "pass"
  end

  -- Do not scan connections from local network IP addresses
  -- (IP address range 127.0.0.1/8)
  if ctx.src.ip.belongs("127.0.0.0/8") then
    return "pass"
  end

  -- 接続はデフォルトでスキャンされます
  return "check"
end
```

使用中のテーブル

1.InterceptContextテーブル

このテーブルは、処理された接続のデータを`intercept_hook`関数に転送するために使用されます。データを使用して、接続をどうするかを決定できます（スキャンせずにスキップ、切断、またはSpIDer Gateによるスキャンのために送信）。Dr.Web Firewall for Linuxではテーブルにデータを入力します。テーブルのデータの中には、`intercept_hook`関数が実行される時点ですでに認識されているものがあります。他の情報（いわゆる「遅延」データ）は、テーブルの該当するフィールドのリクエストに応じて直接評価されます。テーブルには以下のフィールドを含みます。

フィールド	説明	データタイプ
`src`	接続を開始したクライアントのアドレスとポート 例： `if ctx.src.port >= 0 and ctx.src.port <= 1024 then` `  return "pass"` `end`	TcpEndpointテーブル
`dst`	クライアントによって接続が開始されたサーバーのアドレスとポート 例：	TcpEndpointテーブル

フィールド	説明	データタイプ
	`` if ctx.dst.ip.belongs("10.20.30.41/8") then return "reject" end ```	
divert	監視した接続のタイプ。以下のいずれかの値にできます。 • 「output」- 送信接続。 • 「input」- 受信接続。 • 「forward」- トランジット接続。 例： ``` if ctx.divert == "forward" then return "check" end ```	文字列
iface_in	接続元のインターフェース名。 インターフェース名が定義されていない場合は、nil値になります。	文字列
iface_out	接続を初期化するパケットが送信されたインターフェース名。 インターフェース名が定義されていない場合は、nil値になります。	文字列
uid	送信接続を開始したユーザーID。 接続タイプ（divert）が「output」ではない場合、またはUIDを定義できない場合は、nil値になります。	番号
gid	送信接続を開始したグループID。 接続タイプ（divert）が「output」ではない場合、またはGIDを定義できない場合は、nil値になります。	番号
user	送信接続を開始したユーザー名。 接続タイプ（divert）が「output」ではない場合、またはユーザー名を定義できない場合は、nil値になります。	文字列
group	送信接続を開始したグループ名。 接続タイプ（divert）が「output」ではない場合、またはユーザー名を定義できない場合は、nil値になります。	文字列
pid	送信接続を開始したプロセスID。 接続タイプ（divert）が「output」ではない場合、またはPIDを定義できない場合は、nil値になります。	番号
exe_path	送信接続を開始したアプリケーションの実行可能ファイルへのパス。	文字列

フィールド	説明	データタイプ
	接続タイプ(`divert`)が「`output`」ではない場合、または実行可能パスを定義できない場合は、`nil`値になります。	
無効になったメタメソッド: *なし*		

2.TcpEndpointテーブル

このテーブルは、接続ポイント(クライアントまたはサーバー)のアドレスを説明しています。以下のフィールドを含みます。

フィールド	説明	データタイプ
`ip`	IPアドレス	IpAddressテーブル
`port`	ポート番号	番号
無効になったメタメソッド: • `__tostring` - **TcpEndpoint**を文字列に変換する関数。例:「127.0.0.1:443」(IPv4)または「[::1]:80」(IPv6) • `__concat` - **TcpEndpoint**を文字列に追加する関数		

利用可能な補助モジュール

LuaのプログラムスペースでDr.Web for UNIX Internet Gatewaysとやり取りするために、次の特定のモジュールをインポートできます。

モジュール名	機能
drweb	Luaプログラムを起動したDr.Web for UNIX Internet GatewaysコンポーネントとLuaプロシージャの非同期実行の手段のログに、Luaプログラムからのメッセージを記録する機能を提供するモジュール
drweb.lookup	Dr.Web LookupDモジュールを呼び出して外部ソースからデータを要求するためのツールを提供するモジュール

drwebモジュールの内容

1.機能

このモジュールは次の機能を提供します。

1.1.Luaプログラムからのメッセージを Dr.Web for UNIX Internet Gatewaysコンポーネントログに保存するための機能:

- `log(`*<level>*, *<message>*`)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*<level>*レベル(必要なレベルは、*「debug」*、*「info」*、*「notice」*、*「warning」*、*「error」*を使用して定義します)で書き込みます。
- `debug(`*<message>*`)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*DEBUG*レベルで書き込みます。

- `info(<message>)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*INFO*レベルで書き込みます。
- `notice(<message>)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*NOTICE*レベルで書き込みます。
- `warning(<message>)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*WARNING*レベルで書き込みます。
- `error(<message>)`は*<message>*文字列をDr.Web for UNIX Internet Gatewaysログに*ERROR*レベルで書き込みます。

1.2.Luaプロシジャの同期管理のための機能:

- `sleep(<sec.>)`はこのLuaプロシジャインスタンスの実行を指定された秒数で一時停止します。
- `async(<Lua function>[, <argument list>])`は、指定された関数の非同期開始を起動し、指定された引数リストに転送します。`async`関数呼び出しはすぐに完了し、戻り値(**Future**テーブル)を使用すると、*<Lua function>*の結果を取得できます(まだ実行が完了していない場合は、完了するまで待機している可能性があります)。

1.3.IPアドレスをIpAddressテーブルとして表示するための機能:

- `ip(<address>)`は、**IpAddress**テーブルの形式で*<address>*文字列として送信される、IPアドレスを指定します。IPv4またはIPv6アドレスのいずれかを使用できます。

1.4.テキストファイルから外部データをアップロードするには:

- `load_set(<file path>)`は、指定されたテキストファイルのコンテンツから`true`値を含むテーブルを生成し、ファイルから読み取られた文字列はキーとして使用されます。空の文字列と空白文字のみで構成される文字列は無視され、テーブルには含まれません。
- `load_array(<file path>)`は、指定されたテキストファイルのコンテンツから文字列の配列を生成します。空の文字列と空白文字のみで構成される文字列は無視され、配列には含まれません。

2.テーブル

2.1.**Future**テーブルは、`async`関数を使用して関数を実行した後の保留中の結果を表します。テーブルには以下のフィールドを含みます。

フィールド	説明	データタイプ
`wait`	`async`関数を使用して開始した関数の結果を返す関数。関数がまだ実行を完了していない場合は、完了を待って結果を返します。`wait`が呼び出される前に関数が完了した場合、結果はすぐに返されます。開始された関数が失敗した場合、`wait`呼び出しは同じエラーを生成します。	機能
無効になったメタメソッド: *なし*		

2.2 **IpAddress**テーブルはIPアドレスを表します。以下のフィールドを含みます。

フィールド	説明	データタイプ
`belongs`	指定したサブネット(IPアドレス範囲)の**IpAddress**テーブルに保存されているIPアドレスの所属を確認する関数。	機能

フィールド	説明	データタイプ
	「*<IP address>*」または「*<IP address>*/*<mask>*」のような文字列を唯一の引数として受け取ります。ここで、*<IP address>*はホストアドレスまたはネットワークアドレス(「127.0.0.1」など)、*<mask>*はサブネットワークマスク(「255.0.0.0」などのIPアドレスとして、または「8」などの数値形式で指定できます)です。 ブール値を返します。 • `true`は、アドレスが指定されたアドレスの少なくとも1つと等しいか、指定されたサブネット(IPアドレスの範囲)の少なくとも1つに属していることを示します。 • `false` - それ以外の場合。	
無効になったメタメソッド: • `__tostring`は、文字列内の**IpAddress**を変更する関数です。例:「127.0.0.1」(IPv4)または「::1」(IPv6) • `__concat`は、**IpAddress**を文字列に結合する関数です。 • `__eq`は、2つの**IpAddress**が等しいことを確認するための関数です。 • `__band`は、マスクを適用するための関数(例:`dw.ip('192.168.1.2') & dw.ip('255.255.254.0'))`です。		

3.例

3.1.非同期的に開始される手順によって生成される、ログへの出力メッセージ:

```
local dw = require "drweb"

-- この関数は2秒間待機して文字列を返します。
-- 引数として受信されます
function out_msg(message)
 dw.sleep(2)
 return message
end

-- 「メイン」関数
function intercept(ctx)
 -- NOTICEレベルでDr.Web for UNIX Internet Gatewaysログに文字列を出力します
 dw.notice("Intercept function started.")

 -- out_msg関数の2つのコピーを非同期で起動します
 local f1 = dw.async(out_msg, "Hello,")
 local f2 = dw.async(out_msg, " world!")

 -- out_msg関数のコピーの完了を待機中です
 -- out_msgとその結果をログに出力します
 -- Dr.Web for UNIX Internet Gateways ログにデバッグレベルで出力します
 dw.log("debug", f1.wait() .. f2.wait())
end
```

3.2.指定されたスケジュールに従って定期的に開始する手順を作成する:

```
local dw = require "drweb"

-- Futureテーブルをfutureグローバル変数で保存し、
-- Luaのガベージコレクターにより
-- 計画されたタスクが破壊されないようにします
future = dw.async(function()
   while true do
     -- 毎日、次のメッセージがログに表示されます
     dw.sleep(60 * 60 * 24)
     dw.notice("A brand new day began")
   end
end)
```

3.3.文字列のIPアドレスを変更する:

```
local dw = require "drweb"

local ipv4 = dw.ip("127.0.0.1")
local ipv6 = dw.ip("::1")
local mapped = dw.ip("::ffff:127.0.0.1")
```

drweb.lookupモジュールの内容

1.機能

このモジュールは次の機能を提供します。

- `lookup(`*<request>*, *<parameters>*`)`はDr.Web LookupDモジュールから利用できる外部ストレージからデータを要求します。*<request>*パラメータは、Dr.Web LookupD設定内のセクション(文字列*<type>*`@`*<tag>*)に対応している必要があります。*<parameters>*引数はオプションで、リクエストを生成するために使用される置換を表します。これらのパラメータはテーブルとして設定されます。このテーブルのキーと値は文字列でなければなりません。この関数は、リクエストの結果である文字列の配列を返します。
- `check(`*<checked string>*、*<request>*、*<parameters>*)は、Dr.Web LookupDモジュールを介して利用できる外部リポジトリで*<checked string>*が見つかった場合に`true`を返します。引数*<request>*および*<parameters>*は、`lookup`関数の引数と同じです(上記を参照)。*<checked string>*引数は、文字列または`__tostring`メタメソッドを持つテーブル(つまり、文字列にフォーマットできる)であると想定されます。

2.例

データソースLookupD.LDAP.usersから取得されたユーザーリストのログへの出力:

```
local dw = require "drweb"
local dwl = require "drweb.lookup"

-- 「メイン」関数
function intercept(ctx)
 -- NOTICEレベルでDr.Web for UNIX Internet Gatewaysログに文字列を出力します
 dw.notice("Intercept function started.")

 -- リクエスト結果をDr.Web for UNIX Internet Gatewaysログへ出力
 -- 'ldap@users'データソースへ
 for _, s in ipairs(dwl.lookup("ldap@users", {user="username"})) do
   dw.notice("Result for request to 'ldap@users': " .. s)
 end

end
```

Dr.Web ClamD

Dr.Web ClamDコンポーネントは、Sourcefire,Inc.のアンチウイルス製品**Clam AntiVirus**（**ClamAV®**）のコアコンポーネントである**clamd**アンチウイルスデーモンのインターフェースを使用してエミュレーションを実行します。このインターフェースにより、**ClamAV ®**とやり取りできる外部アプリケーションは、アンチウイルススキャンにDr.Web for UNIX Internet Gatewaysを使用できます。

動作原理

このコンポーネントは、ローカルファイルシステムのファイルの内容と、ソケットを介して外部アプリケーションによって送信されたデータのストリームの両方をチェックします。このようなチェックは、外部アプリケーションのリクエストに応じてコンポーネントによって実行されます。さらに、コンポーネントは外部アプリケーションがソケットを介してオープンファイル記述子（ディスクリプタ）を渡したファイルの内容をチェックできます。

渡されたファイル記述子に基づくファイルスキャンは、記述子がローカルのUNIXソケットを介して渡された場合にのみ実行できます。

外部アプリケーションからローカルファイルシステム内のファイルへのパスが提供された場合、コンポーネントはスキャンタスクをDr.Web File Checkerファイルチェッカーコンポーネントに送信します。それ以外の場合、コンポーネントは、ソケット経由で受信したデータをDr.Web Network Checkerに送信します。

デフォルトでは、コンポーネントはDr.Web for UNIX Internet Gatewaysの起動時に自動的に起動されません。コンポーネントの起動を有効にするには、開始パラメータに`Yes`の値を設定し、クライアントアプリケーション用に少なくとも1つの接続ポイントを定義する必要があります。その後、コンポーネントは外部アプリケーションからのファイルまたはデータストリームのスキャンリクエストの待機を開始します。コンポーネントの設定では、外部アプリケーション用に複数の接続ポイントを設定し、必要に応じて各ポイントに異なるスキャン設定を調整できます。

外部アプリケーションは、**clamd**との統合モジュールを装備している場合には、HTTPプロキシサーバー（**Squid**や**HAVP**など）として表すことができます。詳細は、外部アプリケーションとの統合のセクションを参照してください。

検出された脅威はDr.Web for UNIX Internet Gatewaysで駆除することは*できません*。外部アプリケーションにはスキャンの結果のみが通知されます。そのため、検出された脅威はすべて外部アプリケーションで駆除する必要があります。

コマンドライン引数

Dr.Web ClamDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-clamd [<parameters>]
```

Dr.Web ClamDは次のパラメータを処理できます。

パラメータ	説明

`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-clamd --help
```

このコマンドはDr.Web ClamDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じてDr.Web ConfigD設定デーモンによって自動的に起動されます（原則として、OSの起動時）。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-clamd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[ClamD]`セクションで指定されている設定パラメータを使用します。

- コンポーネントパラメータ。
- コンポーネント構成の特別な側面。

コンポーネントパラメータ

セクションには以下のパラメータが含まれています。

`LogLevel` *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、`[Root]`セクションの**`DefaultLogLevel`**パラメータの値が使用されます。 **デフォルト値**：`Notice`
`Log`	コンポーネントのロギング方式。 **デフォルト値**：`Auto`

{log type}	
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-clamd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-clamd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-clamd
Start *{Boolean}*	コンポーネントはDr.Web ConfigD設定デーモンによって起動される必要があります。 このパラメータにYes値を指定すると、設定デーモンはただちにコンポーネントを開始するように指示されます。また、No値を指定すると、設定デーモンはただちにコンポーネントを終了するように指示されます。 **デフォルト値**：No
Endpoint.*<tag>*.**ClamdSocket** *{IP address | UNIX socket}*	*<tag>*という名前の新しい接続ポイントを定義し、ファイルの脅威をスキャンする必要があるクライアントにソケット（IPv4アドレスまたはUNIXソケットのアドレス）を割り当てます。 *1つの<tag>ポイントに指定できるソケットは1つのみです。* **デフォルト値**：*（未設定）*
[Endpoint.*<tag>*.]**DetectSuspicious** *{Boolean}*	ヒューリスティックアナライザによって検出された疑わしいファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：Yes
[Endpoint.*<tag>*.]**DetectAdware** *{Boolean}*	アドウェアを含むファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：Yes
[Endpoint.*<tag>*.]**DetectDialers** *{Boolean}*	ダイアラーを含むファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、こ*

	のパラメータに別の値が指定されていないすべてのポイントに設定されます。 **デフォルト値**：Yes
[Endpoint.*<tag>*.]**DetectJokes** *{Boolean}*	ジョークプログラムを含むファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：No
[Endpoint.*<tag>*.]**DetectRiskware** *{Boolean}*	リスクウェアを含むファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：No
[Endpoint.*<tag>*.]**DetectHacktools** *{Boolean}*	ハッキングツールを含むファイルについて通知します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：No
[Endpoint.*<tag>*.]**ReadTimeout** *{time interval}*	クライアントからのデータを待つ最大タイムアウトを設定します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：5s
[Endpoint.*<tag>*.]**StreamMaxLength** *{size}*	クライアントから受信できるデータの最大サイズを設定します（スキャンするデータをバイトのストリームとして送信するため）。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：25mb

[Endpoint.<*tag*>.]**ScanTimeout** *{time interval}*	クライアントから受信した1つのファイル(またはデータの一部)をスキャンする最大時間を設定します。 *1秒から1時間の範囲の値を指定できます。* Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**:3m
[Endpoint.<*tag*>.]**HeuristicAnalysis** *{On | Off}*	スキャンにヒューリスティック解析を使用するかどうかを示します。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**:On
[Endpoint.<*tag*>.]**PackerMaxLevel** *{integer}*	スキャンできるパックオブジェクトの最大ネストレベルを設定します。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**:8
[Endpoint.<*tag*>.]**ArchiveMaxLevel** *{integer}*	スキャンできるアーカイブの最大ネストレベルを設定します。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**:8
[Endpoint.<*tag*>.]**MailMaxLevel** *{integer}*	スキャンできるメールファイルの最大ネストレベルを設定します。

	0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。 Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：8
[Endpoint.*<tag>*.]**ContainerMaxLevel** *{integer}*	スキャンできるコンテナ内のオブジェクトの最大ネストレベルを設定します。 *0から60までの範囲の値を指定できます。値を0に設定すると、ネストしたオブジェクトはスキャンされません。* Endpoint.*<tag>のプレフィックスが指定されている場合は、パラメータの値が<tag>接続ポイントにのみ設定されることを意味します。それ以外の場合は、このパラメータに別の値が指定されていないすべてのポイントに設定されます。* **デフォルト値**：8
[Endpoint.*<tag>*.]**MaxCompressionRatio** *{integer}*	圧縮/パックされたオブジェクトの最大許容圧縮率（非圧縮サイズと圧縮サイズの比率）を設定します。オブジェクトの比率が制限を超えると、そのオブジェクトはスキャン中にスキップされます。 *圧縮率には2よりも小さい値は指定できません。* **デフォルト値**：500

コンポーネント構成の特別な側面

オプションのEndpoint.*<tag>*プレフィックスが付いているパラメータはグループ化できます。各グループは、クライアントがコンポーネントへの接続に使用する固有の接続*ポイント*（*エンドポイント*）を定義し、固有の*<tag>* IDが割り当てられています。同じグループに属するすべてのスキャンパラメータは、対応する接続ポイントに接続されているクライアントに対してデータがスキャンされる場合にのみ適用可能な設定を定義します。パラメータがEndpoint.*<tag>*のプレフィックスなしで指定されている場合は、すべての接続ポイントの値が設定されます。接続ポイントからパラメータを削除した場合は、このパラメータをプログラムのハードコードされたデフォルト値に戻す代わりに、同じ名前の対応する「親」パラメータの現在の値を使用します（Endpoint.*<tag>*プレフィックスなしで設定）。

ClamdSocketパラメータは、リスニングソケットとこのソケットが対応するグループ（接続ポイント）の両方を定義するため、必ずEndpoint.*<tag>*プレフィックスを付けて指定する必要があります。

例：

外部アプリケーション（サーバー）の2つのグループに対して2つの接続ポイントを設定する必要があるとします。グループをそれぞれ*servers1*と*servers2*と呼びます。また、*servers1*グループのサーバーはUNIXソケットを介して接続できますが、*servers2*グループのサーバーはネットワーク接続を介して接続できます。さらに、ヒューリスティック解析はデフォルトで無効にする必要がありますが、*servers2*グループのサーバーには使用する必要があるとします。次の例は、これを設定する方法を示しています。

1) 設定ファイルで以下のように設定します。

```
[ClamD]
HeuristicAnalysis = Off

[ClamD.Endpoint.servers1]
ClamdSocket = /tmp/srv1.socket

[ClamD.Endpoint.servers2]
ClamdSocket = 127.0.0.1:1234
HeuristicAnalysis = On
```

2) コマンドラインベースの管理ツールDr.Web Ctlの場合：

```
# drweb-ctl cfset ClamD.HeuristicAnalysis Off
# drweb-ctl cfset ClamD.Endpoint -a servers1
# drweb-ctl cfset ClamD.Endpoint -a servers2
# drweb-ctl cfset ClamD.Endpoint.servers1.ClamdSocket /tmp/srv1.socket
# drweb-ctl cfset ClamD.Endpoint.servers2.ClamdSocket 127.0.0.1:1234
# drweb-ctl cfset ClamD.Endpoint.servers2.HeuristicAnalysis On
```

どちらの方法でも効果は同じですが、設定ファイルを編集する場合は、**drweb-configd**コンポーネントにSIGHUP信号を送信して変更した設定を適用する必要があります（これを行うには**drweb-ctl**の`reload`コマンドを発行します）。

外部アプリケーションとの統合

clamadアンチウイルスデーモン（**ClamAV**に含まれています）の1つをエミュレートするこのインターフェースによって、Dr.Web ClamD は**clamd**アンチウイルスデーモンに接続できる外部アプリケーションと通信できます。

以下の表は、アンチウイルススキャンに**clamd**を使用できるアプリケーションの例を示しています。

製品	統合
HTTPサービス	
HTTPプロキシサーバー **Squid**	**clamd**の使い方： インターネットから受信したファイルをスキャンする。 統合要件： 中間コンポーネントとして**squidclamav**または**HAVP**を使用する。 ドキュメントへのリンク：

製品	統合
	Squidのドキュメント：http://www.squid-cache.org/Doc/ **squidclamav**の説明とソースコードファイル：http://squidclamav.darold.net/
アンチウイルススキャンを実行できるHTTPプロキシサーバー **HAVP**	**clamdの使い方**： インターネットから受信したファイルをスキャンする。 **統合要件**： アンチウイルススキャンに **clamd**を使用するよう**HAVP**を設定する `ENABLECLAMD true` `CLAMDSOCKET` *<path_to_clamd_UNIX_socket>* または（UNIXソケットの代わりにTCP接続が使用されている場合）： `ENABLECLAMD true` `CLAMDSERVER` *<IP>* `CLAMDPORT` *<port>* *<path_to_clamd_UNIX_socket>*または *<IP>:<port>*のペアは、Dr.Web ClamDで設定された接続ポイントのソケット（*エンドポイント*）に対応します。 **ドキュメントへのリンク**： **HAVP**のドキュメント：http://www.havp.org/documentation/

clamd アンチウイルスデーモンと同様に、Dr.Web ClamDと直接通信する外部ソフトウェアコンポーネントの設定で、**clamd**に接続するためのアドレスをUNIXソケットまたは設定された接続ポイント（*エンドポイント*）の1つでDr.Web ClamDにリッスンされるTCPソケットへのパスとして指定します。

HAVPのDr.Web ClamDへの接続例：

1. Dr.Web ClamDの設定：

```
[ClamD]
Start = yes

[ClamD.Endpoint.proxy]
ClamdSocket = /var/run/drweb.clamd
```

2. **HAVP**の設定：

```
ENABLECLAMD true
CLAMDSOCKET /var/run/drweb.clamd
```

他のアンチウイルス製品への接続設定（`ENABLE*`パラメータ）は`false`に設定する必要があります。

Dr.Web File Checker

ファイルスキャンコンポーネントDr.Web File Checkerは、ファイルシステムのファイルとディレクトリをスキャンするように設計されています。ファイルシステムオブジェクトをスキャンするためにDr.Web for UNIX Internet Gatewaysの他のコンポーネントによって使用されます。さらに、このコンポーネントは隔離されたファイルを保存しているディレクトリの内容を管理するため、隔離マネージャーとしても機能します。

動作原理

このコンポーネントは、すべてのファイルシステムオブジェクト（ファイル、ディレクトリ、ブートレコード）へのアクセスに使用されます。スーパーユーザー（*root*）権限で起動します。

スキャン済みのすべてのファイルとディレクトリにインデックスを付け、特別なキャッシュで確認されたオブジェクトに関するすべてのデータを保存し、すでにスキャン済みのオブジェクトやそれ以来変更されていないオブジェクトを繰り返しスキャンしないようにします（この場合は、そうしたオブジェクトのスキャンリクエストを受信する際に、キャッシュから取得した以前のスキャン結果が返されます）。

ファイルシステムオブジェクトを確認するリクエストがDr.Web for UNIX Internet Gatewaysのコンポーネントから受信されると、このオブジェクトにスキャンが必要かどうかを確認します。必要な場合は、Dr.Web Scanning Engineに対してスキャンタスクが生成されます。スキャンされたオブジェクトに脅威が含まれている場合、Dr.Web File Checkerはその脅威を検出された脅威レジストリに入れ、駆除（修復、削除、または隔離）します（脅威への対応としてスキャンを開始したクライアントコンポーネントによってこのアクションが指定されている場合）。スキャンは、製品のさまざまなコンポーネントによって開始できます。

スキャン中、ファイルチェックコンポーネントでは、スキャン結果と適用されたアクションがあればそれを詳述するレポートを生成してクライアントコンポーネントに送信します。

標準のスキャン方法とは別に、内部使用には以下の特別な方法があります。

- *「flow」スキャン方法*。このスキャン方法を使用するクライアントコンポーネントは、検出と駆除のパラメータを一度だけ初期化します。これらのパラメータは、このクライアントコンポーネントからのファイルをスキャンするための今後のすべてのリクエストに適用されます。
- *「プロキシ」スキャン方法*。この方法を使用する場合、ファイルチェックコンポーネントは、検出された脅威にアクションを適用せずに、また今後のアクションを許可するために検出された脅威に関するレコードを保存せずに、ファイルをスキャンします。スキャン処理を開始したコンポーネントによって必要なアクションが適用される必要があります。この方法は、Dr.Web ClamDコンポーネントによって使用されます。

Dr.Web Ctlユーティリティのを使用し、Dr.Web Ctlユーティリティの`flowscan`コマンド（**drweb-ctl**コマンドで起動）を使用し、*「フロー」*スキャン方法でファイルをスキャンできます。ただし、通常のオンデマンドスキャンでは、`scan`コマンドを使用することを推奨します。

作業中、ファイルスキャンコンポーネントは脅威のレジストリを保持して隔離を管理するだけでなく、ファイルスキャン全体の統計情報も収集し、最後の1分間、5分間、15分間の、1秒間に確認された平均ファイル数を計算します。

コマンドライン引数

Dr.Web Network Checkerを起動するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-filecheck [<parameters>]
```

Dr.Web File Checkerは次のパラメータを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-filecheck --help
```

このコマンドはDr.Web File Checkerに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのラインから直接起動することはできません。Dr.Web for UNIX Internet Gatewaysの他のコンポーネントからファイルシステムスキャンのリクエストを受信するときに、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理し、必要に応じてファイルをスキャンするには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（**drweb-ctl**コマンドを使用して起動されます）。

Dr.Web File Checkerを使用して任意のファイルまたはディレクトリをスキャンするには、Dr.Web Ctlの`scan`コマンドを使用します。

```
$ drweb-ctl scan <path to file or directory>
```

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-filecheck`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[FileCheck]`セクションで指定されている設定パラメータを使用します。

このセクションは以下のパラメータを保存します。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-filecheck • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-filecheck • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-filecheck
DebugClientIpc *{Boolean}*	詳細なIPCメッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。 **デフォルト値**：No
DebugScan *{Boolean}*	ファイルスキャン中に受信した詳細メッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。 **デフォルト値**：No
DebugFlowScan *{Boolean}*	*「フロー」*メソッドによるファイルスキャンに関する詳細メッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。 **デフォルト値**：No
DebugProxyScan *{Boolean}*	*「プロキシ」*メソッドによるファイルスキャンに関する詳細メッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。通常、このスキャン方法はDr.Web ClamDコンポーネントによって使用されます。 **デフォルト値**：No
DebugCache *{Boolean}*	スキャンのキャッシュされた結果に関する詳細メッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUGの場合など）。 **デフォルト値**：No
MaxCacheSize *{size}*	スキャンしたファイルに関するデータを保存するためのキャッシュの最大許容サイズ。 *0を指定した場合、キャッシュは無効になります。* **デフォルト値**：50mb
RescanInterval *{time interval}*	前回のスキャンの結果がキャッシュ内にある場合にファイルが再スキャンされない期間（保存された情報が最新のものと見なされる期間）。

	パラメータは0秒から1分までの値を指定できます。設定された間隔が1秒未満の場合は遅延がないため、ファイルは任意のリクエストに応じてスキャンされます。 **デフォルト値**：1s
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 *指定する値は、10秒から30日までの範囲内でなければなりません。* **デフォルト値**：30s

Dr.Web Network Checker

ネットワークチェッカーエージェントDr.Web Network Checkerでは、ネットワーク経由で受信したデータをスキャンエンジンでスキャンできる他、分散ファイルでの脅威のスキャンを実行できます。このコンポーネントを使用すると、ネットワークホストを介してデータ(ファイルの内容など)を送受信できるようDr.Web for UNIX Internet Gatewaysがインストールされているネットワークホスト間の接続を調整し、スキャンを実行できます。このコンポーネントは、設定されているすべての利用可能なネットワークホストへのスキャンタスクの自動分配を調整します(ネットワークを介して送受信)。このコンポーネントは、スキャンタスクによって生じる負荷をホスト間に分散させます。リモートホストとの接続が設定されていない場合、コンポーネントはすべてのデータをローカルのDr.Web Scanning Engineにのみ送信します。

このコンポーネントは、ネットワーク接続を介して受信したデータのスキャンに常に使用されます。したがって、コンポーネントが見つからないか使用できない場合、スキャン用にネットワーク接続を介してデータを送信するコンポーネントのパフォーマンスは不正確になります(Dr.Web ICAPD、Dr.Web ClamD)。

このコンポーネントは、Dr.Web File Checkerコンポーネントを置き換えることができないため、ローカルファイルシステムにあるファイルの分散スキャンを整理するようには作られていません。ローカルファイルの分散スキャンを整理するには、Dr.Web MeshDコンポーネントを使用します。

ネットワークを介して転送されるデータのスキャンの負荷が高まると、利用可能なファイル記述子(ファイルディスクリプタ)が枯渇し、スキャンに問題が生じる場合があります。この場合、Dr.Web for UNIX Internet Gatewaysに利用できるファイル記述子の制限数を増やす必要があります。

スキャン時には、SSL/TLSを適用することにより、オープンチャネル経由または保護されたチャネル経由のいずれかでデータを共有できます。安全なHTTPS接続を使用するには、ファイルを共有するホストに適切なSSLサーバー証明書とプライベートキーを提供する必要があります。SSLキーと証明書を生成するには、**openssl**ユーティリティを使用できます。**openssl**ユーティリティを使用して証明書とプライベートキーを生成する方法の例については、のセクション付録E. SSL証明書を生成するを参照してください。

動作原理

このコンポーネントにより、スキャンでローカルファイルシステムのファイルとして表されていないデータをDr.Web Scanning Engineエンジン(ローカルまたはリモートホストにある)に送信できます。このデータは、(Dr.Web ICAPD、Dr.Web ClamD)接続を介してスキャンのデータを送信するコンポーネントによって処理されます。これらのコンポーネントは、ローカルホストにある場合でも、Dr.Web Scanning Engineエンジンへのファイル転送に常にDr.Web Network Checkerを使用することに注意してください。そのため、Dr.Web Network Checkerが利用できない場合、*これらのコンポーネントは正しく機能しません*。

また、Dr.Web Network Checkerは、Dr.Web for UNIX Internet Gatewaysがインストールされているネットワーク(またはその他のDr.Web for UNIXソリューション10.1以降)の特定のノードセットとDr.Web for UNIX Internet Gatewaysを接続することで、ローカルファイルシステムでファイルとして表示されない分散されたチェックデータの有無をまとめて確認できます。これにより、このコンポーネントは、検証でデータを交換する一連のネットワークノードである*スキャンクラスタ*を構築および設定できます(各インスタンスには、Dr.Web Network Checker分散検証エージェントの独自のインスタンスが必要です)。スキャンクラスタに含まれるネットワークの各ノードで、Dr.Web Network Checkerはスキャンデータのタスクの自動分散を実行し、ネットワークを介して、接続が設定されているすべての使用可能なノードに転送します。同時に、リモートノードで使用可能なリソースの量に応じて、データ検証によって発生するノードの負荷分散が実行されます(ロードで使用可能なリソース量の指標とし

て、このノードの核となるDr.Web Scanning Engineのスキャンによって生成される子のスキャンプロセス数が使用されます）。使用されている各ノードでのチェックを待機しているファイルキューの長さも推定されます。

この場合、スキャンクラスタに含まれる任意のネットワークノードは、リモートスキャンにデータを送信するスキャンクライアントの他、検証のために指定されたネットワークノードからデータを受信するスキャンサーバーとして機能します。必要に応じて、ノードがスキャンサーバーまたはスキャンクライアントとしてのみ機能するように分散スキャンエージェントを設定できます。

スキャン用にネットワーク経由で受信したデータは、一時ファイルとしてローカルファイルシステムに保存され、Dr.Web Scanning Engineエンジンに送信されるか、利用できない場合は、スキャンクラスタの他のノードに送信されます。

設定にある**InternalOnly**パラメータを使用すると、Dr.Web Network Checkerの動作モードを管理できます。これは、Dr.Web Network CheckerがDr.Web for UNIX Internet Gatewaysをスキャンクラスタに含めるために使用されるのか、Dr.Web for UNIX Internet Gatewaysローカルコンポーネントのみの内部目的のために使用されるのかを特定します。

ファイルのチェックにDr.Web Network Checkerを使用する独自のコンポーネント（外部アプリケーション）を作成できます（スキャンクラスタのノードに対するスキャンジョブの分配を含む）。そのために、Dr.Web Network Checkerコンポーネントは**Google Protobuf**テクノロジーに基づくカスタムAPIを提供しています。Dr.Web Network Checker APIの他、Dr.Web Network Checkerを使用するクライアントアプリケーションのサンプルコードが、`drweb-netcheck`パッケージの一部として提供されています。

スキャニングクラスタの作成例は、スキャンクラスタを作成するセクションにあります。

コマンドライン引数

Dr.Web Network Checkerを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-netcheck [<parameters>]
```

Dr.Web Network Checkerは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-netcheck --help
```

このコマンドはDr.Web Network Checkerに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接実行することはできません。必要に応じて、Dr.Web ConfigD構成デーモンによって自動的に実行されます（通常はOSの起動時）。コンポーネント設定で、**FixedSocket**パラメータの値が指定されており、**InternalOnly**パラメータが`No`に設定されている場合、エージェントは常に実行されており、指定されたUNIXソケットを介してクライアントに使用可能です。ネットワーク経由でスキャンを開始するには、Dr.Web for UNIX Internet Gateways管理用のDr.Web Ctlコマンドラインツールを使用できます（コマンド**drweb-ctl**で始まるもの）。リモートホストへの接続が設定されていない場合は、ローカルスキャンが開始されます。

Dr.Web Network Checkerを使用して任意のファイルまたはディレクトリをスキャンするには、Dr.Web Ctlツールの`netscan`コマンドを使用します。

```
$ drweb-ctl netscan <path to file or directory>
```

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-netcheck`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの[NetCheck]セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-netcheck • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-netcheck • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-netcheck
FixedSocket *{path to file / address}*	固定されたDr.Web Network Checkerエージェントインスタンスのソケット。 このパラメータが指定されている場合、Dr.Web ConfigD設定デーモンは、このソケットを介してクライアントが使用可能な分散スキャンエージェントの実行中のコンポーネントのコピーが常に存在することを確認します。 **使用可能な値**： • *<path to file>*は、ローカルのUNIXソケットへのパスです。 • *<address>*は、ペア*<IP address>*:*<port>*で示されるネットワークソケットです。 **デフォルト値**：*（未設定）*
InternalOnly *{Boolean}*	コンポーネントの動作モードの管理。 値がYesに設定されている場合、**LoadBalance***設定や**FixedSocket**パラメータの値に関係なく、コンポーネントはDr.Web for UNIX Internet Gatewaysコンポーネントの内部目的にのみ使用され、スキャンクラスタへのDr.Web for UNIX Internet Gatewaysの組み込みや、外部（Dr.Web for UNIX Internet Gatewaysに対して）クライアントアプリケーションの処理には使用されません。 **デフォルト値**：No
RunAsUser	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指

{UID / user name}	定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「`name:`」というプレフィックスを付けて指定します。次に例を示します。**RunAsUser** = `name:123456`。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：`drweb`
IdleTimeLimit *{time interval}*	コンポーネントがアイドル状態を維持できる最大時間。指定された値を超えると、コンポーネントはシャットダウンします。 最小値 - `10s`. **LoadBalanceAllowFrom**パラメータまたは**FixedSocket**パラメータが設定されている場合、この設定は無視されます（時間間隔が経過しても、コンポーネントが動作を終了しません）。 **デフォルト値**：`30s`
LoadBalanceUseSsl *{Boolean}*	安全なSSL/TLS接続を他のホストへの接続に使用するかどうかを決定するインジケーター。 **使用可能な値**： • `Yes` - SSL/TLSを使用するように指示します。 • `No` - SSL/TLSを使用しないように指示します。 *このパラメータが`Yes`に設定されている場合、証明書と対応するプライベートキーを、このホストとこのホストがやり取りするホストに対して指定する必要があります（パラメータ **LoadBalanceSslCertificate**および **LoadBalanceSslKey**）。* **デフォルト値**：`No`
LoadBalanceSslCertificate *{path to file}*	安全なSSL/TLS接続を介して他のホストと通信するためにDr.Web Network Checkerによって使用されるSSL証明書へのパス。 *証明書ファイルとプライベートキーファイル（後述のパラメータで指定されます）は、一致するペアを形成する必要があります。* **デフォルト値**：*（未設定）*
LoadBalanceSslKey *{path to file}*	安全なSSL/TLS接続を介して他のホストと通信するためにDr.Web Network Checkerによって使用されるプライベートキーへのパス。 *証明書ファイルとプライベートキーファイル（上のパラメータで指定されます）は、一致するペアを形成する必要があります。* **デフォルト値**：*（未設定）*
LoadBalanceSslCa *{path}*	信頼済みのルート証明書のリストを含むディレクトリまたはファイルへのパス。これらの証明書には、SSL/TLSプロトコルを介してデータを交換するときにスキャンクラスタ内のエージェントが使用する、証明書の信頼性を証明する証明書が必要です。 *パラメータ値が空の場合、このホストで作業しているDr.Web Network Checkerはやり取りをするエージェントの証明書を認証しま*

	せん。ただし、設定によっては、ホスト上で動作しているエージェントが使用する証明書をこれらのエージェントが認証できます。 **デフォルト値**：*（未設定）*
LoadBalanceSslCrl *{path}*	失効した証明書のシステムリストを含むディレクトリまたはファイルへのパス。 *パラメータ値が指定されていない場合、このホストで実行されている Dr.Web Network Checkerは、やり取りするエージェントの証明書の有効性をチェックしませんが、設定によっては、このホストで実行されているエージェントが使用する証明書の有効性をチェックする場合があります。* **デフォルト値**：*（未設定）*
LoadBalanceServerSocket *{address}*	リモートホストからスキャン用に送信されたファイルを受信するために Dr.Web Network Checkerがこのホスト上で待機しているネットワークソケット（IPアドレスとポート）（スキャンサーバーとして動作できる場合）。 **デフォルト値**：*（未設定）*
LoadBalanceAllowFrom *{IP address}*	Dr.Web Network Checkerがスキャン用のファイルを受信できるリモートネットワークホストのIPアドレス（スキャンサーバーとして）。 *リストをパラメータ値として指定できます。リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ホストアドレス192.168.0.1と10.20.30.45のリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[NetCheck]` ` LoadBalanceAllowFrom = "192.168.0.1", "10.20.30.45"` • 2つの文字列（文字列ごとに1つの値） `[NetCheck]` ` LoadBalanceAllowFrom = 192.168.0.1` ` LoadBalanceAllowFrom = 10.20.30.45` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset NetCheck.LoadBalanceAllowFrom -a 192.168.0.1` `# drweb-ctl cfset NetCheck.LoadBalanceAllowFrom -a 10.20.30.45`

	このパラメータが空の場合、削除されたファイルをスキャンのために受信することはできません（ホストはスキャンサーバーとして動作しません）。 **デフォルト値**：*（未設定）*
LoadBalanceSourceAddress *{IP address}*	リモートスキャン用にファイルを転送するためにホスト上のDr.Web Network Checkerによって使用されるネットワークインターフェースのIPアドレス（ホストがスキャンサーバーとして動作し、複数のネットワークインターフェースを持つ場合）。 *空の値を指定すると、OSカーネルによって自動的に選択されたネットワークインターフェースが使用されます。* **デフォルト値**：*（未設定）*
LoadBalanceTo *{address}*	ホスト上のDr.Web Network Checkerが（ネットワークスキャンクライアントとして）リモートスキャン用のファイルを送信できるリモートホストのソケット（IPアドレスまたはポート）。 *リストをパラメータ値として指定できます。リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ソケット192.168.0.1:1234および10.20.30.45:5678をリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[NetCheck]` ` LoadBalanceTo = "192.168.0.1:1234", "10.20.30.45:5678"` • 2つの文字列（文字列ごとに1つの値） `[NetCheck]` ` LoadBalanceTo = 192.168.0.1:1234` ` LoadBalanceTo = 10.20.30.45:5678` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset NetCheck.LoadBalanceTo -a 192.168.0.1:1234` `# drweb-ctl cfset NetCheck.LoadBalanceTo -a 10.20.30.45:5678` *パラメータ値が空の場合、ローカルファイルをリモートスキャン用に転送できません（ホストはネットワークスキャンクライアントとして動作しません）。* **デフォルト値**：*（未設定）*
LoadBalanceStatusInterval	ワークロードに関する情報を含む次のメッセージをすべてのスキャンクライアントに送信する際にホストによって考慮されるタイムインターバ

{time interval}	ル（**LoadBalanceAllowFrom**パラメータで指定）。 **デフォルト値**：1s
SpoolDir *{path to directory}*	スキャン用にネットワーク経由で送信され、Dr.Web Network Checkerによって受信されたファイルを保存するために使用されるローカルファイルシステムディレクトリ。 **デフォルト値**：/tmp/netcheck
LocalScanPreference *{fractional number}*	ファイル（ローカルファイルまたはネットワーク経由で受信したファイル）をスキャンするスキャンサーバーが選択されたときに考慮される、このホストの相対的な重み（プライオリティー）。ローカルステーションの相対的な重みが、スキャンサーバーとして利用可能なすべてのホストの重みよりも大きい場合、ファイルはローカルでスキャンされます。 最小値 - 1。 **デフォルト値**：1

スキャンクラスタを作成する

このセクションの内容：

- 注意事項
- スキャニングクラスタの作成例
- クラスタノードを設定する
- クラスタの動作を確認する

注意事項

ファイルや他のオブジェクトのスキャン中に分散チェックを実行できるスキャンクラスタを作成するには、各ノードにDr.Web Network Checkerコンポーネントがインストールされた一連のネットワークノードが必要です。クラスタノードにスキャン対象のデータの送受信以外の機能を持たせるには、ノードにスキャンエンジンDr.Web Scanning Engineもインストールする必要があります。そのため、スキャンクラスタのノードを作成するには、次のコンポーネントの最小セット（最低限のもの）をサーバーにインストールする必要があります（ここにリストされているコンポーネントの機能を保証するために自動的にインストールされるDr.Web for UNIX Internet Gatewaysの他のコンポーネントは、スキップされます）。

1. Dr.Web Network Checker（drweb-netcheckパッケージ）は、ノード間のネットワークを提供するコンポーネントです。
2. Dr.Web Scanning Engine（drweb-seパッケージ）は、ネットワーク経由で受信したデータをスキャンするために必要なスキャンエンジンです。場合によっては、このコンポーネントが存在しません。その場合、ノードは他のスキャンクラスタノードに対してチェック対象のデータの送信のみを行います。

*ピアツーピア*ネットワークのスキャニングクラスタを構成するノード、つまり各ノードは、このノードのDr.Web Network Checkerコンポーネントで定義されている設定に応じて、*スキャンクライアント*（他のノードにスキャン用にデータを送信）または*スキャンサーバー*（他のノードからスキャン用にデータを受信）として機能できます。適切に設定すれば、クラスタノードは同時にスキャンクライアントとスキャンサーバーの両方として機能します。

スキャニングクラスタ設定に関連するDr.Web Network Checkerパラメータは、**LoadBalance**で始まる名前を持ちます。

スキャニングクラスタの作成例

下の図に表示されているスキャニングクラスタの構成例を確認してください。

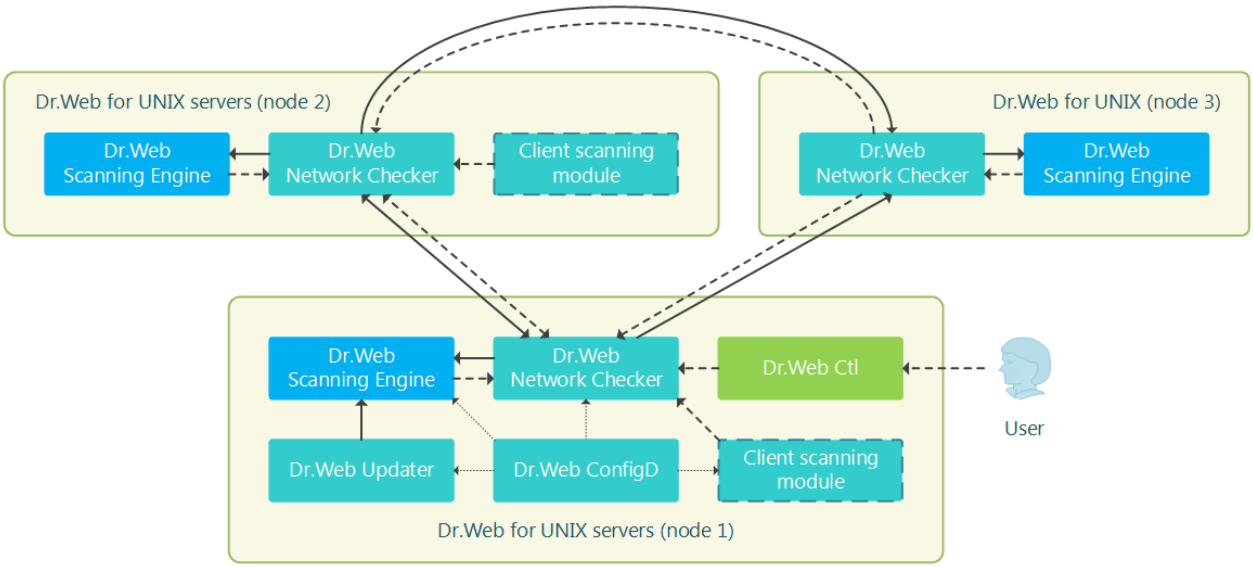

図13. スキャニングクラスタの構造

ここでは、クラスタは3つのノード（図では*ノード1*、*ノード2*、*ノード3*として表示）で構成されると想定します。この場合、ノード1とノード2は、UNIXサーバー用のDr.Webの上位製品がインストールされているサーバーです（Dr.Web for UNIXファイルサーバーまたは Dr.Web for UNIXインターネットゲートウェイなど。製品タイプは不問）。ノード3は、ノード1とノード2から転送されたファイルのスキャンをサポートするためにのみ使用されます。したがって、必要最小限のコンポーネントセット（Dr.Web Network CheckerおよびDr.Web Scanning Engine）のみがインストールされます。ノードの操作性を確保するために自動的にインストールされる他のコンポーネント（Dr.Web ConfigDなど）は図に記載されていません。ノード1と2は、相互間でサーバーおよびスキャンの両方のクライアントとして機能し（スキャンに関連する負荷の相互分散を実行）、ノード3はサーバーとしてのみ機能し、ノード1と2からタスクを受信します。

これらのコンポーネントは、ローカルにインストールされたスキャンエンジンDr.Web Scanning Engineとクラスタパートナーノードの間で分散され、負荷分散に応じてスキャンサーバーとして機能します。

ローカルファイルシステムでファイルとして表されていないデータをスキャンするコンポーネントのみが、検証のクライアントモジュールとして機能できることに注意してください。これは、SpIDer GuardファイルシステムモニターおよびDr.Web File Checkerコンポーネントによるファイルの分散スキャンに使用できるスキャンクラスタを指します。

クラスタノードを設定する

指定したクラスタ構成をカスタマイズするには、すべてのクラスタノードでDr.Web Network Checker設定を変更する必要があります。以下の設定はすべて.iniファイルで指定されます（設定ファイルのフォーマットの説明を参照）。

ノード1

```
[NetCheck]
InternalOnly=No
LoadBalanceUseSsl = No
LoadBalanceServerSocket = <Node 1 IP address>: <Node 1 port>
LoadBalanceAllowFrom = <Node 2 IP address>
LoadBalanceSourceAddress = <Node 1 IP address>
LoadBalanceTo = <Node 2 IP address>: <Node 2 port>
LoadBalanceTo = <Node 3 IP address>: <Node 3 port>
```

ノード2

```
[NetCheck]
InternalOnly=No
LoadBalanceUseSsl = No
LoadBalanceServerSocket = <Node 2 IP address>: <Node 2 port>
LoadBalanceAllowFrom = <Node 1 IP address>
LoadBalanceSourceAddress = <Node 2 IP address>
LoadBalanceTo = <Node 1 IP address>: <Node 1 port>
LoadBalanceTo = <Node 3 IP address>: <Node 3 port>
```

ノード3

```
[NetCheck]
InternalOnly=No
LoadBalanceUseSsl = No
LoadBalanceServerSocket = <Node 3 IP address>: <Node 3 port>
LoadBalanceAllowFrom = <Node 1 IP address>
LoadBalanceAllowFrom = <Node 2 IP address>
```

注意

- 他の（ここに記載されていない）Dr.Web Network Checkerパラメータは変更されません。
- IPアドレスとポート番号は実際のものに変更する必要があります。
- この例では、ノード間のデータ交換におけるSSLの使用は無効になっています。SSLを使用する必要がある場合は、**LoadBalanceUseSsl**パラメータに値`Yes`を設定するとともに、パラメータ**LoadBalanceSslCertificate**、**LoadBalanceSslKey**、**LoadBalanceSslCa**に必要な値を設定する必要があります。

クラスタの動作を確認する

データディストリビューションモードでのクラスタの動作を確認するには、ノード1とノード2で次のコマンドを使用します。

```
$ drweb-ctl netscan <path to file or directory>
```

指定されたコマンドを実行するときは、指定されたディレクトリのファイルをDr.Web Network Checkerでチェックする必要があります。これにより、カスタマイズされたクラスタノードにチェックが分散されます。スキャン前に各ノード

のネットワークチェックの統計を表示するには、次のコマンドを使用してDr.Web Network Checkerの統計の表示を実行します（統計の表示を中断するにはCTRL+Cを押します）。

```
$ drweb-ctl stat -n
```

Dr.Web Scanning Engine

スキャンエンジンDr.Web Scanning Engineではディスクデバイスのファイルやブートレコード（*MBR - マスターブートレコード*、*VBR - ボリュームブートレコード*）内のウイルスやその他の悪意のあるオブジェクトを検索できます。コンポーネントによってスキャンエンジンDr.Web Virus-Finding Engineがメモリに読み込まれて起動される他、エンジンによる脅威の検出に使用するDr.Webウイルスデータベースが読み込まれます。

スキャンエンジンは、他のDr.Web for UNIX Internet Gatewaysコンポーネント（これらはDr.Web File CheckerとDr.Web Network Checker、部分的にはDr.Web MeshD）からスキャンリクエストを受信するサービスとして、デーモンモードで動作します。*Dr.Web Scanning EngineとDr.Web Virus-Finding Engineがないか使用できない場合、このノードではアンチウイルススキャンは実行されません（Dr.Web MeshDコンポーネントを含むDr.Web for UNIX Internet Gatewaysを除く、このコンポーネントではスキャンエンジンサービスを提供するローカルクラウドノードへの接続が設定されます）。*

動作原理

このコンポーネントは、埋め込まれた脅威のDr.Web for UNIX Internet Gatewaysコンポーネントからファイルシステムオブジェクト（ファイルおよびブートディスクレコード）をスキャンするリクエストを受信するサービスとして動作します。また、スキャンタスクをキューに入れ、Dr.Web Virus-Finding Engineスキャンエンジンを使用してリクエストされたオブジェクトをスキャンします。脅威が検出され、スキャンタスクによって脅威を修復するように指示があった場合、このアクションをスキャンされたオブジェクトに適用できるのであれば、スキャンエンジンは修復を試みます。

スキャンエンジン、Dr.Web Virus-Finding Engineスキャンエンジン、ウイルスデータベースは1つの単位を構成しており、分離することはできません。スキャンエンジンはウイルスデータベースをダウンロードし、クロスプラットフォームのスキャンエンジンDr.Web Virus-Finding Engineの動作環境を提供します。ウイルスデータベースとスキャンエンジンは、Dr.Web for UNIX Internet GatewaysにDr.Web Updater更新コンポーネントによって更新されますが、このコンポーネントはスキャンエンジンの一部ではありません。対応するコマンドがユーザーから送信された場合、更新コンポーネントはDr.Web ConfigD設定デーモンによって定期的または強制的に実行されます。さらに、Dr.Web for UNIX Internet Gatewaysが集中管理モードで動作していている場合、ウイルスデータベースとスキャンエンジンの更新は、Dr.Web ES Agentによって実行されます。Dr.Web ES Agentは集中管理サーバーから更新を受け取ります。

Dr.Web Scanning Engineは、設定デーモンDr.Web ConfigDの管理下でも、自律モードでも動作できます。前者の場合、デーモンはエンジンを実行し、アンチウイルスデータベースが最新であることを確認します。後者の場合、エンジンの起動とウイルスデータベースの更新は、エンジンを使用する外部アプリケーションによって実行されます。ファイルのスキャンを要求するスキャンエンジンにリクエストを発行するDr.Web for UNIX Internet Gatewaysコンポーネントは、他の外部アプリケーションと同じインターフェースを使用します。

ユーザーは、ファイルチェックにDr.Web Scanning Engineを使用して独自のコンポーネント（外部アプリケーション）を作成できます。このため、Dr.Web Scanning Engineには**Google Protobuf**に基づく特別なAPIが含まれています。Dr.Web Scanning Engine APIガイドとDr.Web Scanning Engineを使用したクライアントアプリケーションの例を入手するには、Doctor Webパートナーケア部門（https://partners.drweb.com/）にお問い合わせください。

受信したタスクは、優先度（高い、通常、低い）ごとのキューに自動的に分配されます。キューの選択は、タスクを作成したコンポーネントによって異なります。たとえば、ファイルシステムモニターによって作成されたタスクは、応答時間がモニターにとって重要であるため、優先順位が高くなります。スキャンエンジンは、スキャン用に受信したすべてのタスクの数やキューの長さなど、操作に関わる統計を計算します。平均負荷率として、スキャンエンジン

は1秒あたりのキューの平均長を使用します。このレートは、直近1分間、直近5分間、直近15分間の平均です。

Dr.Web Virus-Finding Engineスキャンエンジンは、機械語の命令やその他の実行可能コードの属性に基づいて潜在的に危険なオブジェクトを検出するために設計された、シグネチャ解析（シグネチャベースの脅威の検出）やその他のヒューリスティック解析および動作解析方法をサポートします。

ヒューリスティック解析は信頼性の高い結果を保証できず、次のようなエラーが発生する可能性があります。

- *最初のタイプのエラー*。これらのエラーは、安全なオブジェクトが悪意のあるものとして検出された場合に発生します（誤検知）。
- *2番目のタイプのエラー*。これらのエラーは、悪意のあるオブジェクトが安全であると検出されたときに発生します。

したがって、ヒューリスティックアナライザによって検出されたオブジェクトは*疑わしい*ものとして扱われます。

疑わしいオブジェクトは隔離に移動することをお勧めします。ウイルスデータベースを更新した後、そのようなファイルはシグネチャ解析を使用してスキャンできます。2番目のタイプのエラーを避けるには、ウイルスデータベースを最新の状態に保ってください。

Dr.Web Virus-Finding Engineスキャンエンジンを使用すると、異なるコンテナ内のファイルや圧縮されたオブジェクトまたはオブジェクト（アーカイブ、メールメッセージなど）の両方をスキャンして修復できます。

コマンドライン引数

スキャンエンジンDr.Web Scanning Engineをコマンドラインから実行するには、次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-se <socket> [<parameters>]
```

必須の*<socket>*引数は、クライアントコンポーネントの要求を処理するためにDr.Web Scanning Engineによって使用されるソケットのアドレスを示します。ファイルパス（UNIXソケット）としてのみ設定できます。

Dr.Web Scanning Engineは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None
追加の起動パラメータ（設定ファイルのパラメータと同じものであり、必要に応じて置き換えます）。	

`--EnginePath`	**機能**：Dr.Web Virus-Finding Engineスキャンエンジンのライブラリへのパスを指定します。 **短縮形**：なし **引数**：*<path to the file>* - 使用するライブラリへのフルパス。
`--VirusBaseDir`	**機能**：ウイルスデータベースファイルがあるディレクトリへのパスを指定します。 **短縮形**：なし **引数**： *<path to the directory>* - ウイルスデータベースディレクトリへのパス。
`--TempDir`	**機能**：一時ファイルがあるディレクトリへのパスを指定します。 **短縮形**：なし **引数**： *<path to the directory>* - 一時ファイルを含むディレクトリへのフルパス。
`--Key`	**機能**：キーファイルへのパスを指定します。 **短縮形**：なし **引数**：*<path to the file>* - 使用するキーファイルへのフルパス。
`--MaxForks`	**機能**：スキャン中にDr.Web Scanning Engineによって起動できる子プロセスの最大許容数を指定します。 **短縮形**：なし **引数**：*<number>* - 子プロセスの最大許容数。
`--WatchdogInterval`	**説明**：Dr.Web Scanning Engineが子プロセスが動作可能かどうかをチェックし、応答を停止したプロセスを停止する頻度を指定します。 **短縮形**：なし **引数**：*<time interval>* - 子プロセスをチェックする頻度。
`--Shelltrace`	**機能**：シェルトレースをオンにします（Dr.Web Virus-Finding Engineによって実行されたファイルスキャンの詳細情報をログに記録します）。 **短縮形**：なし **引数**： None
`--LogLevel`	**説明**：操作中にDr.Web Scanning Engineによって実行されるロギングのレベルを設定します。 **短縮形**：なし **引数**：*<logging level>*。可能な値は次のとおり。 • `DEBUG` - 最も詳細なログレベル。すべてのメッセージとデバッグ情報が登録されます。 • `INFO` - すべてのメッセージが登録されます。 • `NOTICE` - すべてのエラーメッセージ、警告、通知が登録されます。 • `WARNING` - すべてのエラーメッセージと警告が登録されます。 • `ERROR` - エラーメッセージのみが登録されます。
`--Log`	**説明**：コンポーネントメッセージのロギングの方法を指定します。 **短縮形**：なし **引数**：*<log type>*。可能な値は次のとおり。 • `Stderr[:ShowTimestamp]` - メッセージは標準エラーストリームの*stderr*に出力されます。 追加オプション`ShowTimestamp`は、すべてのメッセージにタイムスタンプを追加するように指示します。 • `Syslog[:<facility>]` - メッセージはシステムロギングサービス**syslog**に送信されます。

	追加オプション *<facility>*は、**syslog**のメッセージ登録レベルを指定するために使用します。次の値を使用できます。 ○ DAEMON - デーモンのメッセージ。 ○ USER - ユーザープロセスのメッセージ。 ○ MAIL - メールプログラムのメッセージ。 ○ LOCAL0 - ローカルプロセス0のメッセージ。 ... ○ LOCAL7 - ローカルプロセス7のメッセージ。 • *<path>* - すべてのメッセージが登録されているファイルへのパス。 **例**： --Log /var/opt/drweb.com/log/se.log --Log Stderr:ShowTimestamp --Log Syslog:DAEMON

例:

```
$ /opt/drweb.com/bin/drweb-se /tmp/drweb.ipc/.se --MaxForks=5
```

このコマンドはDr.Web Scanning Engineスキャンエンジンのインスタンスを起動し、そのインスタンスに対してクライアントコンポーネントとのインタラクション用の/tmp/drweb.ipc/.se UNIXソケットを作成し、ファイルのスキャン中に開始する子スキャンプロセスを5つ以下にするように指示します。

スタートアップノート

必要に応じて、Dr.Web Scanning Engineスキャンエンジンの任意の数のインスタンスを起動できます。インスタンスは、(Dr.Web for UNIX Internet Gatewaysコンポーネントだけでなく)クライアントアプリケーションにスキャンサービスを提供します。その場合、コンポーネントの設定で**FixedSocketPath**パラメータの値が指定されていると、スキャンエンジンの1つのインスタンスがDr.Web ConfigD設定デーモンによって常に実行され、このUNIXソケットを介してクライアントから利用可能になります。コマンドラインから直接起動したスキャンエンジンのインスタンスは、設定デーモンが実行中であっても、設定デーモンへの接続を確立せずに自律モードで動作します。コンポーネントの動作を管理し、必要に応じてファイルをスキャンするには、Dr.Web for UNIX Internet GatewaysのDr.Web Ctlコマンドラインベースの管理ツールを使用できます(**drweb-ctl**コマンドを使用して起動されます)。

Dr.Web Scanning Engineを使用して任意のファイルまたはディレクトリをスキャンするには、Dr.Web Ctlツールのrawscanコマンドを使用します。

```
$ drweb-ctl rawscan <path to file or directory>
```

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** 1 drweb-se

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイル

このセクションは以下のパラメータを保存します。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-se • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-se • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-se
FixedSocketPath *{path to file}*	Dr.Web Scanning Engineスキャンエンジンの固定インスタンスのUNIXソケットへのパス。 このパラメータが指定されている場合、Dr.Web ConfigD設定デーモンは、このソケットを介してクライアントが使用可能なスキャンエンジンのコピーが常に実行されていることを確認します。 **デフォルト値**：*（未設定）*
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 最小値 - 10s. *__FixedSocketPath__パラメータが設定されている場合、この設定は無視されます（時間間隔が経過しても、コンポーネントが動作を終了しません）。* **デフォルト値**：30s
MaxForks *{integer}*	同時に実行できる、Dr.Web Scanning Engineによって実行される子プロセスの最大許容数。 **Default value**: *使用可能なCPUコアの数の2倍が自動的に使用されます。算出された数が4未満の場合は4になります。*
BufferedIo *{On | Off}*	ファイルをスキャンするときは、バッファ付き入出力（I/O）を使用します。 **FreeBSD**および**GNU/Linux** *OSでバッファ付きI/Oを使用すると、低速ディスクでのファイルスキャン速度を上げることができます。* **デフォルト値**：Off
WatchdogInterval *{time interval}*	応答を停止したプロセス（「watchdog」）を検出するために、子プロセスが動作可能かどうかをDr.Web Scanning Engineがチェックする頻度です。 **デフォルト値**：1.5秒

Dr.Web Updater

更新コンポーネントDr.Web Updaterは、Doctor Web更新サーバーからウイルスデータベースおよびDr.Web Virus-Finding Engineスキャンエンジンの利用可能な更新をすべて受信し、更新をDr.Web for UNIX製品コンポーネントのローカルクラウドと同期（Dr.Web MeshDが製品に含まれている場合はそれを経由）させることを目的に設計されています。

Dr.Web for UNIX Internet Gatewaysが集中管理モードで動作している場合、更新は集中管理サーバー（Dr.Web Enterprise Serverなど）から受信されます。その場合、すべての更新はDr.Web ES Agent経由でサーバーから受信され、Dr.Web Updaterは更新のダウンロードには使用されません（Dr.Web for UNIX製品のローカルクラウドと更新が同期されることもありません）。

動作原理

このコンポーネントは、Doctor Web更新サーバーへの接続を確立して、ウイルスデータベースとDr.Web Virus-Finding Engineスキャンエンジンの更新と、Webリソースカテゴリーのデータベースの更新を確認することを目的に設計されています。利用可能な更新ゾーンを構成するサーバーのリストは、特別なファイルに保存されます（ファイルは改変を防ぐために署名されています）。Doctor Web更新サーバーへの接続では、基本認証とダイジェスト認証のみがサポートされています。

Dr.Web for UNIX Internet Gatewaysが集中管理サーバーに接続されていない場合、またはモバイルモードでサーバーに接続されている場合、Dr.Web UpdaterはDr.Web ConfigD設定デーモンによって自動的に起動されます。起動は、設定で指定された周期で実行されます。適切なコマンドをユーザーから受け取った場合、コンポーネントは設定デーモンによって起動することもできます（スケジュールされていない更新）。

サーバー上で利用可能になった更新は、*<var_dir>*/cacheディレクトリ（**GNU/Linux**の場合は`var/opt/drweb.com/cache`）にダウンロードされ、その後Dr.Web for UNIX Internet Gatewaysの作業ディレクトリに移されます。

デフォルトでは、更新はすべてDr.Webの全製品に共通の更新ゾーンから実行されます。更新ゾーンに含まれる、デフォルトで使用されるサーバーのリストは、***DrlDir**パラメータで定義されたディレクトリにあるファイルで指定され、更新タイプ別にグループ化されています（ウイルスデータベースおよびスキャンエンジンと、Webリソースカテゴリーのデータベース）。これらのファイルは、更新されたコンポーネント（ウイルスデータベース、スキャンエンジン、アンチスパムコンポーネント）によってグループ化されています。ユーザーのリクエストに応じて（更新タイプごとに）特別な更新ゾーンを作成できます。これが、***CustomDrlDir**パラメータで指定されたディレクトリにある、別のファイル（デフォルト名は`custom.drl`）で指定されているサーバーリストです。この場合、更新コンポーネントは、デフォルトゾーンのサーバーを使用せずに、これらのサーバーからのみ更新を受信します。

特別な更新ゾーンを使用しない場合は、コンポーネント設定で該当するパラメータの***CustomDrlDir**値を削除します。

サーバーリストを含むファイルの中身は署名されているため、ファイルを変更することはできません。更新サーバーの特別なリストを作成する必要がある場合は、テクニカルサポートにお問い合わせください。

コンポーネントは、ユーザーのリクエストに応じて、更新のロールバックに備えて、更新したファイルをバックアップできます。バックアップしたファイルの場所と詳細レベルを設定で指定できます。更新をロールバックするには、

Dr.Web CtlDr.Web Ctlコマンドラインからソリューションを管理するためのDr.Web for UNIX Internet Gatewaysのコマンドラインツールを使用します（**drweb-ctl**コマンドで実行されます）。

Dr.Web for UNIX Internet GatewaysがDr.Web for UNIX製品のローカルクラウドに接続されていて、集中管理サーバーに接続されていない場合、Dr.Web Updaterコンポーネントはクラウドホストによって受信された更新の同期にも使用されます。つまり、更新サーバーが受信した更新をクラウドに送信し、クラウドから更新を受信することになるため、Dr.Web更新サーバーの総負荷を軽減できます。このオプションはコンポーネント 設定で有効または無効にできます。

コマンドライン引数

Dr.Web Updaterを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-update [<parameters>]
```

Dr.Web Updaterは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-update --help
```

このコマンドはDr.Web Updaterに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理し、ウイルスデータベースとスキャンエンジンを更新するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-update`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの[Update]セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log {log type}	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-update • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-update • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-update
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。**RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
UpdateInterval *{time interval}*	Dr.Web更新サーバーで更新を確認する頻度。これは、（自動または手動で開始した）更新サーバーへの接続が成功してから次に更新の実行を試みるまでの時間間隔です。 **デフォルト値**：30m
RetryInterval *{time interval}*	前回の試行が失敗した場合に、更新サーバーを使用して更新の実行を再試行する頻度。 パラメータには、1mから30mまでの値を指定できます。 **デフォルト値**：3m
MaxRetries *{integer}*	前回の試行が失敗した場合に、更新サーバーを使用して（**RetryInterval**で指定された頻度で）更新の実行を繰り返し試みる回数。 *値が0に設定されている場合、試行は繰り返されません（次の更新は**UpdateInterval**で指定された時間間隔の後に実行されます）。* **デフォルト値**：3

Proxy *{connection string}*	Dr.Web更新サーバーへの接続時にUpdaterコンポーネント（Dr.Web Updater）が使用するプロキシサーバーに接続するためのパラメータを保存します（外部サーバーへの直接接続がネットワークのセキュリティポリシーによって禁止されている場合など）。 パラメータ値が指定されていない場合、プロキシサーバーは使用されません。 **使用可能な値：** *<connection string>*は、プロキシサーバーの接続文字列です。文字列のフォーマット（URL）は以下のとおりです。 `[`*<protocol>*`://][`*<user>*`:`*<password>*`@]`*<host>*`:`*<port>* 各パラメータは次のとおりです。 • *<protocol>*は、使用されるプロトコルタイプです（現在のバージョンでは、`http`のみが使用可能です）。 • *<user>*は、プロキシサーバーに接続するためのユーザー名です。 • *<password>*は、プロキシサーバーに接続するためのパスワードです。 • *<host>*は、プロキシのホストアドレスです（IPアドレスまたはドメイン名、つまりFQDN）。 • *<port>*は使用するポートです。 URLの*<protocol>*および*<user>:<password>*の部分がない場合があります。プロキシサーバーのアドレス*<host>:<port>*は必須です。 *ユーザー名またはパスワードに「@」、「%」、「:」の文字が含まれている場合、これらの文字はそれぞれ「%40」、「%25」、「%3A」の16進コードに変更する必要があります。* **例：** 1. 設定ファイルでの設定。 • ポート*123*を使用した*proxyhost.company.org*でホストされているプロキシサーバーへの接続： **Proxy** = proxyhost.company.org:123 • ポート*3336*を使用し、パスワードが「passw」のユーザー「legaluser」としてHTTPプロトコルを経由した*10.26.127.0*でホストされているプロキシサーバーへの接続： **Proxy** = http:// legaluser:passw@10.26.127.0:3336 • ポート*3336*、ユーザー名「user@company.com」、パスワード「passw%123:」を使用した*10.26.127.0*でホストされているプロキシサーバーへの接続： **Proxy** = user%40company.com:passw%25123%3A@10.26.127.0:3336 2. コマンド**drweb-ctl** `cfset`を使用して同じ値を設定する場合： `# drweb-ctl cfset Update.Proxy proxyhost.company.org:123` `# drweb-ctl cfset Update.Proxy http://legaluser:passw@10.26.127.0:3336` `# drweb-ctl cfset Update.Proxy user%40company.com:passw%25123%3A@10.26.127.0:3336`

	デフォルト値：*（未設定）*
ExcludedFiles *{file name}*	Dr.Web Updaterコンポーネントによって更新されないファイルの名前を定義します。 *リストをパラメータ値として指定できます。リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：以下のファイルをリストに追加します。`123.vdb`および`456.dws`。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[Update]` `ExcludedFiles = "123.vdb", "456.dws"` • 2つの文字列（文字列ごとに1つの値） `[Update]` `ExcludedFiles = 123.vdb` `ExcludedFiles = 456.dws` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset Update.ExcludedFiles -a 123.vdb` `# drweb-ctl cfset Update.ExcludedFiles -a 456.dws` **デフォルト値**：`drweb32.lst`
NetworkTimeout *{time interval}*	更新プロセス中にUpdaterコンポーネントのネットワーク関連の動作に課されるタイムアウト時間。 このパラメータは、接続が一時的に切断されたときに使用されます。タイムアウトが切れる前に接続が再度確立された場合、中断された更新プロセスが続行されます。 *`75s`を超えるタイムアウト値を指定した場合は、TCPタイムアウトによって接続が閉じられるため効力を持ちません。許容される最小値は`5s`です。* **デフォルト値**：`60s`
BaseDrlDir *{path to directory}*	標準的な更新ゾーンの更新サーバーへの接続に使用されるファイルを含むディレクトリへのパスを定義します。更新コンポーネントがウイルスデータベースおよびスキャンエンジンを更新するために使用されます。 **デフォルト値**：*<var_dir>*`/drl/bases` • **GNU/Linux**の場合：`/var/opt/drweb.com/drl/bases` • **FreeBSD**の場合：`/var/drweb.com/drl/bases`
BaseCustomDrlDir *{path to directory}*	特別な「カスタマイズされた」更新ゾーンへの接続に使用されるファイルを含むディレクトリへのパスを定義します。ウイルスデータベースとスキャンエンジンを更新するために使用されます。

	パラメータで定義されたディレクトリ内に、空でない署名付きサーバーリストファイル（.drlファイル）がある場合、更新はこれらのサーバーからのみ実行され、主要なゾーンサーバー（上記参照）はウイルスデータベースおよびスキャンエンジンの更新には使用されません。 **デフォルト値：** *<var_dir>*/custom-drl/bases • **GNU/Linuxの場合**：/var/opt/drweb.com/custom-drl/bases • **FreeBSDの場合**：/var/drweb.com/custom-drl/bases
BaseUpdateEnabled *{Boolean}*	ウイルスデータベースとスキャンエンジンの更新が許可されているかどうかを示すインジケーター。 **使用可能な値：** • Yes - 更新は許可されており、実行されます。 • No - 更新は許可されていないため実行されません。 **デフォルト値**：Yes
VersionDir *{path to directory}*	サーバーへの接続に使用されるファイルを含むディレクトリへのパスを定義します。Dr.Web for UNIX Internet Gatewaysのバージョンの更新に使用されます。 **デフォルト値：** *<var_dir>*/drl/version • **GNU/Linuxの場合**：/var/opt/drweb.com/drl/version • **FreeBSDの場合**： /var/drweb.com/drl/version
VersionUpdateEnabled *{Boolean}*	Dr.Web for UNIX Internet Gatewaysコンポーネントのバージョンの更新が許可されているかどうかを示すインジケーター。 **使用可能な値：** • Yes - 更新は許可されており、実行されます。 • No - 更新は許可されていないため実行されません。 **デフォルト値**：Yes
DwsCustomDrlPath *{path to file}*	特別な更新ゾーンのサーバーのリストを含む署名付きファイルへのパス。Webリソースカテゴリーのデータベースを更新するために使用されます。 *パラメータが空ではなく、指定されたファイルが存在する場合、更新にはサーバーのみが使用されます。リストのメインファイル（上を参照）は無視されます。パラメータによって認識されたファイルが空の場合、更新の試行は失敗します。* **デフォルト値：** *<var_dir>*/drl/dws/custom.drl • For **GNU/Linux**: /var/opt/drweb.com/drl/dws/custom.drl • **FreeBSDの場合**：/var/drweb.com/drl/dws/custom.drl
DwsDrlDir *{path to directory}*	標準的な更新ゾーンのサーバーに接続するためのファイルを含むディレクトリへのパスを定義します。Webリソースカテゴリーのデータベースを更新するために使用されます。 **デフォルト値：** *<var_dir>*/drl/dws • **GNU/Linuxの場合**：/var/opt/drweb.com/drl/dws • **FreeBSDの場合**：/var/drweb.com/drl/dws

DwsCustomDrlDir *{path to directory}*	特別な「カスタマイズされた」更新ゾーンのサーバーに接続するためのファイルを含むディレクトリへのパスを定義します。Webリソースカテゴリーのデータベースを更新するために使用されます。 *パラメータで定義されたディレクトリ内に、空でない署名付きサーバーリストファイル（.drlファイル）がある場合、更新はこれらのサーバーからのみ実行され、主要なゾーンサーバー（上を参照）はWebリソースカテゴリーのデータベースを更新するためには使用されません。* **デフォルト値**：*<var_dir>*/custom-drl/dws • **GNU/Linux**の場合：/var/opt/drweb.com/custom-drl/dws • **FreeBSD**の場合：/var/drweb.com/custom-drl/dws
DwsUpdateEnabled *{Boolean}*	Webリソースカテゴリーのデータベースの更新が許可されているかどうかを示すインジケーター。 **使用可能な値**： • Yes - 更新は許可されており、実行されます。 • No - 更新は許可されていないため実行されません。 **デフォルト値**：Yes
AntispamDrlDir *{path to directory}*	パラメータは使用されません。 **デフォルト値**：*<var_dir>*/drl/antispam • **GNU/Linux**の場合：/var/opt/drweb.com/drl/antispam • **FreeBSD**の場合：/var/drweb.com/drl/antispam
AntispamCustomDrlDir *{path to directory}*	パラメータは使用されません。 **デフォルト値**：*<var_dir>*/custom-drl/antispam • **GNU/Linux**の場合：/var/opt/drweb.com/custom-drl/antispam • **FreeBSD**の場合：/var/drweb.com/custom-drl/antispam
AntispamUpdateEnabled *{Boolean}*	パラメータは使用されません。 **デフォルト値**：No
BackupDir *{path to directory}*	ロールバックに備えて、更新済みファイルの旧バージョンが保存されているディレクトリへのパス。更新するたびに、更新したファイルのみがバックアップされます。 **デフォルト値**：/tmp/update-backup
MaxBackups *{integer}*	更新済みファイルの以前のバージョンの最大保存数。この数を超えると、最も古いコピーが次回の更新時に削除されます。 *パラメータ値がゼロの場合、以前のバージョンのファイルは保存されません。* **デフォルト値**：0
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。

	最小値 - 10s. *コンポーネントは、スケジュールによる次の更新時または明示的なコマンド* **drweb-ctl** update [--local-cloud] *で起動されます。更新が完了すると、指定された時間間隔の間、待機します。新しいリクエストがない場合（*`UseLocalCloud = Yes`*の場合のクラウドとのインタラクションを含む）は、次の更新を試行するまでシャットダウンします。* **デフォルト値**：30s
UseLocalCloud *{Boolean}*	Dr.Web更新サーバーに加えて、Dr.Web MeshDコンポーネント経由でDr.Web for UNIX製品のローカルクラウドと連携して、更新を同期します（クラウドに更新を送信し、クラウドから更新を取得します）。 • No-更新にはDr.Web更新サーバーのみ使用します。クラウドとの更新の同期は無効ですが、**drweb-ctl** update --local-cloudコマンドを使用して明示的に実行できます。 • Yes - ホスト上の更新をローカルクラウドと同期します（利用可能な更新がある場合はクラウドから更新を取得し、ホスト上の更新のほうが新しい場合はクラウドに更新を送信します）。 **デフォルト値**：Yes

Dr.Web ES Agent

アンチウイルス集中管理エージェントDr.Web ES Agentは、Dr.Web for UNIX Internet Gatewaysを集中管理サーバー（Dr.Web Enterprise Serverなど）に接続するように設計されています。

Dr.Web for UNIX Internet Gatewaysが集中管理サーバーDr.Web ES Agentに接続されている場合、とライセンスキーファイルは、集中管理サーバーに保存されているキーファイルに従って同期されます。さらに、Dr.Web ES Agentウイルスイベントに関する統計、実行中のコンポーネントのリストとステータスを集中管理サーバーに送信します。

またDr.Web ES Agentは、更新コンポーネントDr.Web Updaterを経由せずに、接続されている集中管理サーバーから直接Dr.Web for UNIX Internet Gatewaysのウイルスデータベースを更新します。

動作原理

Dr.Web ES Agentは、集中管理サーバー（Dr.Web Enterprise Serverなど）への接続を確立します。これにより、ネットワーク管理者はネットワーク内で共通のセキュリティポリシーを実装し、特にすべてのネットワークステーションとサーバーに対して同じスキャン設定と脅威検出への対応を設定できます。さらに、集中管理サーバーは最新のウイルスデータベース、の最新バージョンのリポジトリを保存するため、ネットワークの内部更新サーバーの役割も果たします（この場合、更新はDr.Web ES Agentで実行され、Dr.Web Updaterは使用されません）。

Dr.Web ES Agentを集中管理サーバーに接続する際、エージェントはプログラムコンポーネントとライセンスキーファイルの最新の設定を確実に受信し、管理対象コンポーネントに適用するためにDr.Web ConfigD設定デーモンに送信されるようにします。さらに、コンポーネントはステーションのファイルシステムオブジェクトをスキャンするタスク（スケジュールされたタスクを含む）も受信します。

Dr.Web ES Agentは、検出された脅威と適用されたアクションに関するサーバー統計を収集して送信します。

Dr.Web ES Agentを集中管理サーバーに接続するには、ホスト（集中管理サーバーでは「ステーション」）のパスワードとID、認証のためにサーバーによって使用されるパブリック暗号化キーファイルが必要です。ステーションIDの代わりに、ステーションが含まれるメインおよび課金プライングループのIDを指定できます。必要なIDとパブリックキーファイルについては、アンチウイルスネットワークの管理者に問い合わせてください。

さらに、このオプションが集中管理サーバーで許可されている場合は、ホストを保護されたサーバー（「ワークステーション」）に「新規端末」として接続できます。この場合、管理者が接続リクエストを確認した後に、集中管理サーバーでは自動的にIDとパスワードを生成し、今後の接続のためにそれをエージェントに送信します。

コマンドライン引数

Dr.Web ES Agentを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-esagent [<parameters>]
```

Dr.Web ES Agentは次のオプションを処理できます。

パラメータ	説明

`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-esagent --help
```

このコマンドはDr.Web ES Agentに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。Dr.Web ConfigD設定デーモンによって自動的に起動します（OSの起動時）。コンポーネントの動作を管理し、Dr.Web for UNIX Internet Gatewaysを集中管理サーバーに接続するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-esagent`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[ESAgent]`セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、`[Root]`セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：`Notice`
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：`Auto`
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*`/bin/drweb-esagent` • **GNU/Linux**の場合：`/opt/drweb.com/bin/drweb-esagent`

	• **FreeBSDの場合**：`/usr/local/libexec/drweb.com/bin/drweb-esagent`
DebugIpc *{Boolean}*	詳細なIPCメッセージがデバッグレベルでログに含まれるかどうかを示します（**LogLevel** = DEBUGの場合）（Dr.Web ES AgentとDr.Web ConfigD設定デーモンのインタラクション）。 **デフォルト値**：No
MobileMode *{On / Off / Auto}*	集中管理サーバーに接続したときにモバイルモードで動作するDr.Web for UNIX Internet Gatewaysの機能を決定します。 **使用可能な値**： • On - 集中管理サーバーで許可される場合はモバイルモードを使用するように指示します（つまり、Dr.Web Updaterを介してDoctor Webの更新サーバーから更新を実行します）。 • Off - モバイルモードを使用せずに集中管理モードで動作を継続するように指示します（更新は常に集中管理サーバーから受信されます）。 • Auto - 集中管理サーバーで許可される場合はモバイルモードを使用し、どの接続が利用でき、品質が高いかに応じて、Dr.Web Updaterを介したDoctor Webの更新サーバーと集中管理サーバーの両方から更新を実行します。 *このパラメータの動作はサーバーの権限に依存することに注意してください。モバイルモードが使用するサーバーで許可されていない場合、このパラメータは無効です。* **デフォルト値**：Auto
Discovery *{On / Off}*	エージェントが、集中管理サーバーに組み込まれているネットワークインスペクターからの*discovery*リクエストを受信できるかどうかを示します（*discovery*リクエストは、アンチウイルスネットワークの構造と状態を確認するためにインスペクターによって使用されます）。 **使用可能な値**： • On - エージェントは*discovery*リクエストを受信して処理できます。 • Off - エージェントは*discovery*リクエストを受信して処理できません。 *このパラメータは、集中管理サーバーの設定よりも優先順位が高いことに注意してください。パラメータ値がOffに設定されている場合、このオプションがサーバーで有効になっていてもエージェントはdiscoveryリクエストを受信しません。* **デフォルト値**：On
UpdatePlatform *{platform name}*	エージェントが集中管理サーバーからスキャンエンジンの更新を受信するように指示します。スキャンエンジンは指定されたプラットフォーム向けに開発されました。*プラットフォーム名*はプラットフォーム名を含む文字列です。 **使用可能な値**： • **GNU/Linux**：unix-linux-32、unix-linux-64、unix-linux-mips • **FreeBSD**：unix-freebsd-32、unix-freebsd-64 • **Darwin**：unix-darwin-32、unix-darwin-64

	 パラメータ値は確実に変更が必要な場合のみ変更することを強くお勧めします。 **デフォルト値**：*現在使用されているプラットフォームに依存*

Dr.Web HTTPD

Dr.Web HTTPDは、HTTP経由で（たとえばWebブラウザ経由で）Dr.Web for UNIX Internet Gatewaysとローカルおよびリモートで対話するためのインフラストラクチャを提供します。コンポーネントには、Dr.Web for UNIX Internet Gatewaysを管理するためのインターフェース（Dr.Web HTTPDのインストールに加えて、Webブラウザ経由で製品を管理するためのWebインターフェースのファイルを含む個別のパッケージもインストールする必要があります）と、**Mozilla Firefox**ブラウザと**Google Chrome**ブラウザの拡張機能であるDr.Web Link Checkerコンポーネント（お使いのブラウザで使用可能な場合には個別にインストールされます）によって使用されるサービスインターフェースの2つがあります。

Dr.WebのWebインターフェースを介してDr.Web for UNIX Internet Gatewaysを管理するだけでなく、Dr.Web HTTPDのコマンドインターフェース（HTTP API）を直接使用し、HTTPSを介してDr.Web for UNIX Internet Gatewaysのコンポーネントと対話することもできます。この機能により、Dr.Web for UNIX Internet Gatewaysを管理するための独自のインターフェースを作成できます。

Dr.Web HTTPDが提供するHTTP APIの詳細については、該当箇所を参照してください。

安全なHTTPS接続を使用するには、Dr.Web HTTPDに適切なSSLサーバー証明書とプライベートキーを提供する必要があります。デフォルトでは、インストール中にDr.Web HTTPDのSSLサーバー証明書とSSLプライベートキーが自動的に生成されますが、必要に応じて独自の証明書とキーを生成することもできます。また、Dr.Web HTTPDによって信頼されている認証局証明書で署名されたユーザーの個人用認証証明書を、Dr.Web HTTPDに接続するときの自動クライアント認証に使用することもできます。

SSLキーと証明書を生成するには、**openssl**ユーティリティを使用できます。**openssl**ユーティリティを使用して証明書とプライベートキーを生成する方法の例については、のセクション付録E. SSL証明書を生成するを参照してください。

動作原理

Dr.Web HTTPDは、Dr.Web for UNIX Internet Gatewaysの動作を管理するためのWebサーバーです。Dr.Web HTTPDがあれば、外部Webサーバー（**Apache HTTP Server**や**Nginx**など）や**Webmin**などの管理サービスを使用せずに済みます。さらに、コンポーネントは、同じホスト上にあるそのようなサーバーやサービスと同時に機能することができ、それらの動作を妨げることはありません。

Dr.Web HTTPDサーバーは、HTTPおよびHTTPSプロトコルを介して設定で指定されたソケットで受信したリクエストを処理します。このため、このサーバーは、Webサーバーと同じホスト上で動作しているときに、それらのサーバーと競合することはありません。セキュアHTTPSプロトコルはDr.Web for UNIX Internet Gatewaysの管理に使用され、HTTPプロトコルはDr.Web Link Checkerブラウザ拡張リクエスト処理（ブラウザに個別にインストールされます）に使用されます。

> Dr.Web管理WebインターフェースとDr.Web Link Checkerプラグインをインストールすることは、Dr.Web for UNIX Internet Gatewaysを正しく機能させるための必須事項ではありません。これらはない場合もあります。対応するブロックが破線で囲まれているのはこのためです。

Dr.Web HTTPDコンポーネントは、Dr.Web for UNIX Internet Gateways Dr.Web ConfigD設定デーモン、ファイルスキャン用のDr.Web File Checkerコンポーネント、およびその他のコンポーネントにコマンドを送信します。これらのコマンドは、提供されるHTTP APIを介して受信されたコマンド（管理Webインターフェース経由で作成されたコマンドやDr.Web Link Checkerブラウザプラグインからのリクエストとして作成されたコマンドを含む）に基づいています。

Dr.Web HTTPDを使用するDr.Web for UNIX Internet Gatewaysの管理WebインターフェースがDr.Web for UNIX Internet Gatewaysに含まれている場合は、セクションにその説明が記載されています。

Dr.Web管理WebインターフェースがDr.Web for UNIX Internet Gatewaysに含まれていない場合は、Dr.Web HTTPDによるHTTP APIを対話に使用する任意の外部管理インターフェースを接続できます（セクションHTTP APIの説明で説明しています）。

コマンドライン引数

Dr.Web HTTPDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-httpd [<options>]
```

Dr.Web HTTPDは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-httpd --help
```

このコマンドはDr.Web HTTPDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じてDr.Web ConfigD構成デーモンによって自動的に起動されます（通常はOSの起動時）。コンポーネントが実行されていてWebインターフェースがインストールされている場合は、標準のWebブラウザを使用して、Webインターフェースが提供されているアドレスにHTTPS経由でアクセスするだけで、Dr.Web for UNIX Internet Gatewaysのコンポーネントを管理できます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-httpd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの[HTTPD]セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-httpd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-httpd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-httpd
Start *{Boolean}*	コンポーネントはDr.Web ConfigD設定デーモンによって起動される必要があります。 このパラメータにYes値を指定すると、設定デーモンはただちにコンポーネントを開始するように指示されます。また、No値を指定すると、設定デーモンはただちにコンポーネントを終了するように指示されます。 **デフォルト値**：*製品管理インターフェースがインストールされているかどうかによって異なります。*
AdminListen *{address, ...}*	Dr.Web HTTPDが、管理者権限を持つクライアントからの（HTTP経由での）接続をリッスン（待ち受け）しているネットワークソケット（すべてのネットワークソケットは*<IPアドレス>:<ポート>*で構成されます）のリスト。これらのソケットは、管理Webインターフェース（Webインターフェースがインストールされている場合）への接続とHTTP APIへのアクセスの両方に使用されます。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ソケット192.168.0.1:1234および10.20.30.45:5678をリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[HTTPD]` ` AdminListen = "192.168.0.1:1234",` `"10.20.30.45:5678"`

	• 2つの文字列（文字列ごとに1つの値） `[HTTPD]` ` AdminListen = 192.168.0.1:1234` ` AdminListen = 10.20.30.45:5678` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset HTTPD.AdminListen -a 192.168.0.1:1234` `# drweb-ctl cfset HTTPD.AdminListen -a 10.20.30.45:5678` *値が指定されていない場合、HTTP APIとWebインターフェース（インストールされている場合）を使用することはできません。* **デフォルト値**：`127.0.0.1:4443`
`PublicListen` *{address, …}*	Dr.Web HTTPDが、権限が制限されているクライアント、たとえばブラウザにロードされたWebページに悪意のあるオブジェクトがないかスキャンするDr.Web Link Checkerブラウザプラグイン（このプラグインがインストールされている場合）などからの（HTTP経由での）接続をリッスン（待ち受け）しているネットワークソケット（すべてのネットワークソケットは *<IPアドレス>:<ポート>*で構成されます）のリスト。 *リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ソケット192.168.0.1:1234および10.20.30.45:5678をリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 `[HTTPD]` ` PublicListen = "192.168.0.1:1234", "10.20.30.45:5678"` • 2つの文字列（文字列ごとに1つの値） `[HTTPD]` ` PublicListen = 192.168.0.1:1234` ` PublicListen = 10.20.30.45:5678` 2. コマンド**drweb-ctl** `cfset`を使用して値を追加します。 `# drweb-ctl cfset HTTPD.PublicListen -a 192.168.0.1:1234` `# drweb-ctl cfset HTTPD.PublicListen -a 10.20.30.45:5678` *値が指定されていない場合、Dr.Web Link Checkerブラウザプラグインを使用することはできません。*

	これらのアドレス（ソケット）では、HTTP APIのすべてのコマンドにアクセスしたり、管理Webインターフェースにアクセスしたりすることはできません。 **デフォルト値**：*（未設定）*
AdminSslCertificate *{path to file}*	Webインターフェースサーバーが管理ソケットへの接続を確立するクライアントとHTTPS経由で通信するために使用するサーバー証明書ファイルへのパス。 *このファイルは、コンポーネントのインストール中に自動的に生成されます。* *証明書ファイルとプライベートキーファイル（後述のパラメータで指定されます）は、一致するペアを形成する必要があります。* **デフォルト値**：*<etc_dir>*/certs/serv.crt • **GNU/Linux**の場合：/etc/opt/drweb.com/certs/serv.crt • **FreeBSD**の場合：/usr/local/etc/drweb.com/certs/serv.crt
AdminSslKey *{path to file}*	Webインターフェースサーバーが管理ソケットへの接続を確立するクライアントとHTTPS経由で通信するために使用するプライベートキーファイルへのパス。 *このファイルは、コンポーネントのインストール中に自動的に生成されます。* *証明書ファイル（前述のパラメータで指定されます）とプライベートキーファイルは、一致するペアを形成する必要があります。* **デフォルト値**：*<etc_dir>*/certs/serv.key • **GNU/Linux**の場合：/etc/opt/drweb.com/certs/serv.key • **FreeBSD**の場合：/usr/local/etc/drweb.com/certs/serv.key
AdminSslCA *{path to file}*	HTTPS経由で管理ソケットに接続しているクライアントから提供された証明書をチェックするための信頼できるCA証明書として機能する証明書ファイルへのパス。 クライアントの証明書がこのパラメータで設定された証明書で署名されている場合、このクライアントは認証のためにログイン／パスワードのペアを入力する必要はありません。また、このパラメータで設定された証明書で署名されているクライアント証明書を使用するクライアントでは、ログイン／パスワードによる認証は禁止されます。 *この証明書ベースの認証に成功したクライアントは、常にスーパーユーザー（ルート）として扱われます。* **デフォルト値**：*（未設定）*
WebconsoleRoot *{path to directory}*	管理Webインターフェースがインストールされている場合にその管理Webインターフェースによって使用されるファイルを格納するディレクトリ（**Apache HTTPサーバー**のhtdocsディレクトリ相当）へのパス。 **デフォルト値**：*<opt_dir>*/share/drweb-httpd/webconsole • **GNU/Linux**の場合：/opt/drweb.com/share/drweb-httpd/webconsole

	• **FreeBSD**の場合：`/usr/local/libexec/drweb.com/bin/drweb-httpd/webconsole`
LinkcheckerRoot *{path to directory}*	Dr.Web Link Checker Webブラウザプラグインによって使用されるファイルを格納するディレクトリへのパス。 **デフォルト値**：*<opt_dir>*`/share/drweb-httpd/linkchecker` • **GNU/Linux**の場合：`/opt/drweb.com/share/drweb-httpd/linkchecker` • **FreeBSD**の場合：`/usr/local/libexec/drweb.com/share/drweb-httpd/linkchecker`
AccessLogPath *{path to file}*	クライアントからWebインターフェースサーバーへのすべてのHTTP/HTTPSリクエストが登録されるファイルへのパス。 *指定しない場合、HTTP/HTTPSリクエストはファイルに記録されません。* **デフォルト値**：*（未設定）*

HTTP APIの説明

このセクションの内容：

1.概要

HTTP APIは、HTTPプロトコルを介してDr.Web for UNIX Internet Gatewaysを制御および管理する手段として提供されます（セキュリティを確保するために、HTTPSプロトコルが使用されます）。

APIは、HTTPプロトコルの標準メソッドである`GET`と`POST`を使用します。APIは、HTTPプロトコルのバージョン1.0を使用します。HTTP APIのコマンドのパラメータまたはそのようなコマンドの実行結果を表すデータは、HTTPリクエストの本文でJSONオブジェクトとしてテキスト形式で送信されます（コマンドの説明で特に明記されていない場合）。HTTP POSTリクエストの本文でJSONオブジェクトを送信する場合は、このようなリクエストのヘッダーの`Content-Type:`フィールドで`application/json`をその値として指定する必要があります。

HTTPリクエストに対するHTTPレスポンスのフォーマット

- HTTP APIコマンドを処理した結果、後で個別に説明するいくつかの特別な例を除き、レスポンスでJSONオブジェクト（行われたリクエストに固有のJSONオブジェクトか、このAPI呼び出しの処理中に例外が発生した場合はError JSONオブジェクト）が返されます。

- レスポンスとして送信されたJSONオブジェクトにArrayタイプのフィールドがあり、この配列に要素が1つも含まれていない場合、このフィールドはサーバーからのレスポンスから省略されます。
- サーバーレスポンスの`Content-Type`:ヘッダーフィールドは、個別に説明するいくつかの例を除き`application/json`に設定されます。
- クライアントがHTTP APIにないコマンドを呼び出した場合、EC_UNEXPECTED_MESSAGEを値として保持する*code*という名前のフィールドを持つ`Error` JSONオブジェクトを含むレスポンスが返されます。
- SCSが使用される場合(下記を参照)、レスポンスには*SCS cookie*が含まれます。

JSONオブジェクト内の文字列のエンコード

- 文字列は、UTF-8エンコーディング(BOMなし)で送信されます。ASCII表の一部ではない記号は、送信JSON文字列内で`\uXXXX`のようなシーケンスでエスケープされませんが、UTF-8エンコードで送信されます。
- 受信JSONオブジェクトの文字列には、UTF-8でエンコードされた記号と`\uXXXX`のようなエスケープシーケンスの両方を含めることができます。

データ転送に関する一般的な制限

- 本文内のJSONオブジェクトを待つリクエスト(`POST`メソッドによるリクエスト)では、RFC 7159の観点から正しいシンボルであれば、どんなシンボルでも許可されます。
- *REST*アーキテクチャスタイルの要件に従って情報を送信する`GET`メソッドによるリクエストの場合は、RFC 1945に抵触しないシンボルであれば、どんなシンボルでもURIに含めることができます。
- RFC 1945に抵触しないシンボルは、リクエストの他の場所(ヘッダー、本文)に含めることもできます。

2.ユーザー認証と承認

クライアントがHTTPコマンドにアクセスするには、サーバーによってこのようなアクセスが許可されている必要があります。次の2つの認証/承認方法が用意されています。

1. RFC 6896に準拠したSCS(Secure Cookie Sessions for HTTP)を使用する。
2. Dr.Web HTTPDが信頼できるCAの証明書と見なす特別な証明書で署名されたクライアントのSSL証明書を使用する。この場合、クライアントは、認証を受けるためにrootの認証情報を正しく入力したかのように扱われます(X.509クライアント証明書が使用されます)

Secure Cookie Sessionが使用される場合、クライアントがAPIの使用を許可されていることを確認する*Cookie*(以下では単に*SCS cookie*と呼びます)が従来の方法、つまりHTTPリクエスト/レスポンスのヘッダー(`Cookie`:はクライアントからサーバーに送信されるCookieを指定し、`Set-Cookie`:はサーバーからクライアントに送信されるCookieを指定します)で送信されます。クライアントの証明書に基づく認証が使用される場合は、*SCS-cookie*がクライアントから送信されても無視されます。

Secure Cookie Sessionを使用する場合、クライアントとDr.Web HTTPD間の対話は、クライアントがAPIの`login`コマンドを送信し、APIをさらに使用するための認証を受けることから始まる必要があります。この場合、クライアントは認証に成功すると、クライアントが認証されたことを確認する*SCS cookie*を受け取ります。クライアント証明書に基づく認証が使用される場合は、`login`コマンドを送信する必要はありません。また、このコマンドを使用して認証を受けようとすると、認証が拒否されレスポンスでエラー(Error JSONオブジェクト)が返されます。

2.1.ログインとパスワードを指定する(SCS)

認証コマンドの`login`および`logout`は、SCSによる認証方法が使用される場合にのみ使用されます。そうでない場合は、これら2つのコマンドを呼び出そうとすると、呼び出しが拒否され、エラーが返されます(より正確には、エラーコードを含むErrorオブジェクトが返されます)。

ユーザー認証および承認コマンド:

APIコマンド	説明
`login`	アクション: 指定されたユーザー名とパスワードに基づいてクライアントを認証し、HTTP APIのコマンドを使用することをクライアントに許可します。認証が成功すると、*SCS-cookie*が返されます。 **URI**: `/api/10.2/login` **HTTPメソッド**: POST **入力パラメータ**: AuthOptionsオブジェクト **正常に実行された結果**: 空のオブジェクト、*SCS cookie*
`logout`	アクション: 提供された*SCS cookie*を取り消します。その後、取り消された*SCS cookie*を含むHTTP API呼び出しへのレスポンスとして、「EC_NOT_AUTHORIZED」エラーコードを含む`Error`オブジェクトが返されます。 **URI**: `/api/10.2/logout` **HTTPメソッド**: GET **入力パラメータ**: *SCS cookie* **正常に実行された結果**: 空のオブジェクト
`whoami`	アクション: 認証されたユーザーの名前を調べます。 **URI**: `/api/10.2/whoami` **HTTPメソッド**: GET **入力パラメータ**: (*SCS cookie*) * **正常に実行された結果**: whoamiオブジェクト、(*SCS cookie*)

*) *SCS cookie*は、SCSによる認証が使用される場合にのみ送受信する必要があるため、これ以降は括弧に入れて表記します。

使用中のオブジェクトの説明:

1)**`AuthOptions`** - 完全なHTTP APIを使用するために認証および承認される必要があるユーザーのログインデータを含むオブジェクト:

```
{
 "user": string, //User name
```

```
 "password": string //User's password
}
```

2) **whoami** - HTTP APIを使用することを許可されたユーザーの名前を含むオブジェクト:

```
{
 "whoami" :
 {
 "user": string //User name
 }
}
```

3) **Error** - 発生したエラーに関する情報を含むオブジェクト:

```
{
 "error" :
 {
 "code" : string, //A string specifying an error code that looks like
EC_XXX
 *"what": string //Description of the error
 }
}
```

リクエストの処理中にエラーが発生した場合にHTTP APIコマンドへのレスポンスとして返されるError JSONオブジェクトには、数値のエラーコードではなく、Dr.Web for UNIX Internet Gatewaysのコンポーネントによって使用される内部文字列型コードを含む*code*フィールドがあります。このコードは、*EC_XXX*のような文字列です。対応する数値コードを見つけてエラーに関する詳細情報を入手するには、「エラーの内部カタログ」を参照してください。

2.2. 個人証明書を使用する認証

個人証明書を使用して認証する場合、`login`および`logout`コマンドは使用されません。代わりに、HTTPS接続を確立するときに個人認証証明書が使用されます。個人証明書は、Dr.Web HTTPD設定で信頼できるものとして指定された認証局証明書で署名されます。このメカニズムを使用すると、受信したリクエストは*root*ユーザーの代わりに実行されるリクエストと見なされます。

個人証明書を使用する認証の場合:

1. 認証局証明書で署名された個人証明書を作成します。
2. Dr.Web HTTPD設定(パラメータ**AdminSslCA**)で、個人証明書に署名する認証局証明書へのパスを指定します。
3. Dr.Web HTTPDに接続するたびに、署名付き証明書を使用します。

必要に応じて、の付録E. SSL証明書を生成するセクションを参照してください。

3.Dr.Web for UNIX Internet Gatewaysを管理する

設定パラメータの現在の値を取得し、設定を変更するためのAPIコマンド:

APIコマンド	説明
設定を管理するコマンド	
get_lexmap	アクション: 現在の設定(ここではパラメータの語彙マップと呼ばれます)のパラメータ値を取得します。 **URI**: /api/10.2/get_lexmap **HTTPメソッド**: GET **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: LexMapsオブジェクト、(*SCS-cookie*)
set_lexmap	アクション: 現在の設定の指定されたパラメータを設定または(デフォルトに)リセットします(パラメータの「語彙マップ」として送信されます)。 **URI**: /api/10.2/set_lexmap **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*)、LexMapオブジェクト **正常に実行された結果**: SetOptionResultオブジェクト、(*SCS cookie*)
コマンドの更新	
start_update	アクション: 更新を起動します。 **URI**: /api/10.2/start_update **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: StartUpdateオブジェクト、(*SCS-cookie*)
stop_update	アクション: アクティブな更新プロセスを停止します。 **URI**: /api/10.2/stop_update **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: 空のオブジェクト、(*SCS cookie*)
ライセンス管理コマンド	

APIコマンド	説明
`install_license`	**アクション**: 指定されたキーファイルをインストールします。 **URI:** `/api/10.2/install_license` **HTTPメソッド**: POST **入力パラメータ**: (*SCS-cookie*)、キーファイル本体(またはキーファイルを含むアーカイブ) **正常に実行された結果**: 空のオブジェクト、(*SCS cookie*)
集中管理サーバーに接続するためのコマンド	
`esconnect`	**アクション**: 集中管理モードを有効にします。 **URI:** `/api/10.2/esconnect` **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*)、ESConnectionオブジェクト **正常に実行された結果**: 空のオブジェクト、(*SCS cookie*)
`esdisconnect`	**アクション**: 集中管理モードをオフにします。 **URI:** `/api/10.2/esdisconnect` **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: 空のオブジェクト、(*SCS cookie*)

製品のコンポーネントの設定が返され、いわゆる*語彙マップ*、つまり一連のパラメータと値のペアとして設定されます。LexMapsオブジェクトには、3つの語彙マップである以下の3つのフィールドが常に含まれます(つまり、3つのLexMapsオブジェクトが含まれます)。

- *active* - 設定パラメータのアクティブ値、つまり現在有効な値のマップ。
- *hardcoded* - 値が欠落しているか許可されていない場合に設定パラメータに割り当てられるデフォルト値のマップ。
- *master* - クライアントによって設定された設定パラメータの値のマップ。

> `get_lexmap`コマンドは、実際にインストールされて実行されているコンポーネントだけでなく、Dr.Web for UNIX Internet Gatewaysに含めることができるすべてのコンポーネントについて、常に3つすべての設定パラメータ値を返します。

使用されるJSONオブジェクトの説明:

1) **LexMaps** - パラメータ値のアクティブ、デフォルト、およびユーザー設定の語彙マップを含むオブジェクト

```
{
 "active":LexMap, //Active (current) values of configuration parameters
 "hardcoded":LexMap, //Default values of configuration parameters
 "master":LexMap //Configuration parameter values set
 //by the user
}
```

これらの各フィールドは、次にLexOptionオブジェクトの配列が格納されるLexMapオブジェクトです。

2) **LexMap** - パラメータの語彙マップを含むオブジェクト

```
{
 "option":LexOption[] //Array of configuration options
}
```

3) **LexOption** - 単一のパラメータまたは設定のセクション(パラメータのグループ)を含むオブジェクト

```
{
 "key": string, //Name of the option (configuration parameter/section)
 *"value":LexValue, //If this option is a single parameter
 *"map":LexMap //If this option is a section
}
```

LexOptionオブジェクトは、Dr.Web for UNIX Internet Gatewaysの設定のセクションまたは単一のパラメータを表します。このオブジェクトには、常にセクションの名前または単一のパラメータの名前に対応する*key*フィールドがあります。これに加えて、このオブジェクトが表すもの(単一のパラメータまたはセクション)に応じて、*value*フィールド(単一のパラメータを表す場合)または*map*フィールド(セクションを表す場合)もあります。セクションもまた、LexMapタイプのオブジェクトです。一方、単一のパラメータの値は、パラメータの値を文字列形式で指定する*item*フィールドを含むLexValueタイプのオブジェクトです。

4) **LexValue** - パラメータに割り当てられた値の配列を含むオブジェクト

```
{
 "item": string[] //Array of parameter values
}
```

set_lexmapコマンドは、その入力としてLexMapオブジェクトを受け取ります。これには、値を新しい値に変更するか、デフォルトにリセットするすべてのパラメータを含める必要があります。デフォルト値にリセットするパラメータには、*value*フィールドを含めないでください。ユーザーがset_lexmapコマンドで指定した語彙マップに記載されていないパラメータは変更されません。set_lexmapコマンドは、その実行の結果として、コマンドで指定されたすべてのパラメータの変更結果を含むSetOptionResultオブジェクトを返します。

5) **SetOptionResult** - *item*フィールドにパラメータの変更結果の配列を含むオブジェクト

```
{
 "item":SetOptionResultItem[] //Array of results
}
```

このオブジェクトには、コマンドで指定されたすべてのパラメータの変更結果を表すSetOptionResultItemオブジェクトの配列が含まれています。

6) **SetOptionResultItem** - あるパラメータの値を変更することに関する情報を含むオブジェクト。

```
{
 "option": string, //Name of the parameter
```

```
 "result": string, //Result of changing the value (error code)
 *"lower_limit": string, //The lowest permitted value
 *"upper_limit": string //The highest permitted value
}
```

*option*フィールドには、アクションが適用されたパラメータの名前が含まれており、*result*フィールドには、このパラメータの値を変更しようとした結果が含まれています。このフィールドは、指定された値を割り当てようとしたときに発生したエラーに関する文字列タイプのコードです。新しい値がパラメータに正常に割り当てられた場合、このフィールドには`EC_OK`が含まれます。エラーの場合（このフィールドが`EC_OK`に等しくない場合）、このオブジェクトには、このパラメータの最大許容値と最小許容値を保持する*lower_limit*フィールドと*upper_limit*フィールドをオプションで含めることができます。

7）**StartUpdate**オブジェクトには、開始された更新プロセスに関するデータが含まれています

```
{
 "start_update":
  {
    "attempt_id" : number //Identifier of a launched updating process
  }
}
```

8）**ESConnection**オブジェクトには、開始された更新プロセスに関するデータが含まれています

```
{
 *"server": string,    //<Host address>:<port>
 "public_key": string, //Base64 server key
 *"newbie": boolean,   //False by default
 *"login": string,     //User name
 *"password": string   //Password
}
```

パラメータ*login*と*password*は、*newbie* = `true`の場合にのみ指定されます。

4.オブジェクトのスキャン

オブジェクトをスキャンするためのAPIコマンド:

APIコマンド	説明
ファイルスキャン（Dr.Web Network Checkerコンポーネント呼び出しを使用）	
`scan_request`	アクション：必要なパラメータを使用してファイルをスキャンする接続（*endpoint*）の順序。 **URI:** `/api/10.2/scan_request` **HTTP**メソッド：POST 入力パラメータ：（*SCS cookie*）、ScanOptionsオブジェクト 正常に実行された結果：ScanEndpointオブジェクト、（*SCS-cookie*）

APIコマンド	説明
`scan_endpoint`	アクション：作成された*endpoint*接続でのデータスキャン（ファイル本体など）の起動。 **URI:** `/api/10.2/scan_endpoint/<endpoint>` **HTTPメソッド**：POST **入力パラメータ**：（*SCS-cookie*）、検証可能なデータ **正常に実行された結果**：ScanReportオブジェクト、（*SCS-cookie*）
Webページカテゴリーデータベースの URLの確認	
`categorize_address`	アクション：サーバーアドレスがWebページカテゴリーデータベースにリストされているかどうかを確認します。 **URI:** `/api/10.2/categorize_address` **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*）、CategorizeAddressオブジェクト **正常に実行された結果**：CategorizedResourceオブジェクト、（*SCS-cookie*）
`categorize_url`	アクション：URLがWebページカテゴリーデータベースにリストされているかどうかを確認します。 **URI:** `/api/10.2/categorize_url` **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*）、CategorizeURLオブジェクト **正常に実行された結果**：CategorizedResourceオブジェクト、（*SCS-cookie*）
Dr.Web Cloudクラウドの URLとファイルのスキャン	
`cloud_check_url`	アクション：Dr.Web Cloudで、WebページカテゴリーデータベースにリストされているURLに関するデータを確認します。 **URI:** `/api/10.2/cloud_check_url` **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*）、CheckUrlRequestオブジェクト **正常に実行された結果**：UrlResultオブジェクト、（*SCS-cookie*）
`cloud_check_file`	アクション：Dr.Web Cloudで、ファイルの悪質性に関するデータを確認します。 **URI:** `/api/10.2/cloud_check_file` **HTTPメソッド**：POST

APIコマンド	説明
	入力パラメータ：（*SCS cookie*）、CheckFileRequestオブジェクト **正常に実行された結果**：FileResultオブジェクト、（*SCS-cookie*）

使用されるJSONオブジェクトの説明：

1）**ScanOptions**は、ファイルスキャン用の*エンドポイント*を作成するために使用されるパラメータを含むオブジェクトです。

```
{
 "scan_timeout_ms": number, //A time-out to scan one file, in ms
 "cure": boolean, //Apply cure to infected file
 "heuristic_analysis": boolean, //Use heuristic analysis
 "packer_max_level": number, //Max packed object nesting level
 "archive_max_level": number, //Max archive nesting level
 "mail_max_level": number, //Max email object nesting level
 "container_max_level": number, //Max container nesting level
 "max_compression_ratio": number //Max archive compression value
}
```

2）**ScanEndpoint**は、ファイルスキャン用に作成された*エンドポイント*に関するデータを含むオブジェクトです。

```
{
 "endpoint": string //Unique identifier of the created endpoint
}
```

オブジェクト本体で返される*endpoint*文字列は、scan_endpointコマンド（URIの一部）でファイルスキャンを開始するために使用されます。

3）**ScanReport**は、脅威が検出されたファイルに関する情報を含むオブジェクト。

```
{
 "scan_endpoint": string, //File system object that contains the threat
 "size": number, //Size (in bytes) of the file that contains the threat
 *"virus":VirusInfo[], //List of details about the found
 //threats
 *"error": string, //An error message
 "heuristic_analysis": bool //Flag that shows whether heuristic
 //analysis was used
}
```

* *virus*フィールドと*error*フィールドは、スキャン中に脅威が検出されず、エラーが発生しなかった場合には、存在しない可能性があります。scan_endpointを呼び出すため、*scan_endpoint*フィールドでは、Dr.Web Network Checkerコンポーネントによってローカルサーバーファイルシステムに作成され、スキャンに関するデータを含み、scan_endpointリクエストの本文で送信される一時ファイルを必ず指定します。

4）**VirusInfo**は、検出された脅威に関する情報を含むオブジェクトです。

```
{
 "type": string, //Type of the detected threat
 "name": string  //Name of the threat
}
```

5）**CategorizeAddress**は、検証されるURLに関するデータを含むオブジェクトです。

```
{
 "address" : string //IP address for verification
}
```

6）**CategorizeURL**は、検証されるURLに関するデータを含むオブジェクトです。

```
{
 "url" : string //URL for verification
}
```

7）**CategorizedResource**は、Webページの検証結果（ページがWebページカテゴリーデータベースにリストされているかどうかなど）を含むオブジェクトです。

```
{
 *"category": string,     //The main category
 "categories": string[], //List of all categories the resource falls
under
 *"legal_url": string     //URL of the copyright owner
}
```

*legal_url*フィールドは、リソースカテゴリーが「UC_OWNERS_NOTICE」の場合にのみ存在します。*category*フィールドは存在しない場合があります。リソースカテゴリーは文字列「UC_XXX」です。

- UC_INFECTION_SOURCE - 脅威のソース。
- UC_NOT_RECOMMENDED - アクセスが推奨されないWebページ。
- UC_ADULT_CONTENT - アダルトコンテンツ。
- UC_VIOLENCE - 暴力。
- UC_WEAPONS - 武器。
- UC_GAMBLING - ギャンブル。
- UC_DRUGS - 薬物。
- UC_OBSCENE_LANGUAGE - 卑猥な表現。
- UC_CHATS - チャット。
- UC_TERRORISM - テロリズム。
- UC_FREE_EMAIL - 無料メールサービス。
- UC_SOCIAL_NETWORKS - ソーシャルネットワーク。
- UC_ONLINE_GAMES - オンラインゲーム。
- UC_ANONYMIZERS - アノニマイザー。
- UC_CRYPTOCURRENCY_MINING_POOL - 仮想通貨マイニングプール。
- UC_JOBS - 求人検索リソース。
- UC_OWNERS_NOTICEは、著作権者からの申し立てによってリストに登録されたWebページ。

8）**CheckUrlRequest**は、Dr.Web Cloudによる検証の対象となるURLに関するデータを含むオブジェクトです。

```
{
 "url": string //URL to be verified
}
```

9）**UrlResult**は、Dr.Web CloudによるURL検証の結果に関するデータを含むオブジェクトです。

```
{
 "result": string,       //Error code EC_XXX
 "categories": string[] //List of all categories the resource falls under
(see above)
}
```

リソースが正常に検証されると、エラーコード「EC_OK」が返されます。リソースが既知のどのカテゴリーにも属していない場合、*categories*配列は空になります。

10）**CheckFileRequest**は、Dr.Web Cloudによって検証されるURLに関するデータを含むオブジェクトです。

```
{
 "sha1": string,        //SHA1 checksum of the scanned file, a hexadecimal
number
 "path": string,        //Path to a file
 "size": number,        //File size, in bytes
 *"source_url": string //URL of the file source (optional field)
}
```

11）**FileResult**は、Dr.Web CloudによるURL検証の結果に関するデータを含むオブジェクトです。

```
{
 "result": string,    //Error code EC_XXX
 *"virus":VirusInfo //Information on a threat
}
```

ファイルが正常に検証されると、エラーコード「EC_OK」が返されます。脅威が検出されない場合、*virus*フィールドはありません。

12）**VirusInfo**は、検出された脅威に関する情報を含むオブジェクトです。

```
{
 "type": string, //Type of the detected threat
 "name": string //Name of the threat
}
```

*type*フィールド（脅威タイプ）は文字列「SE_XXX」です。

- SE_KNOWN_VIRUSは既知のウイルスです。
- SE_VIRUS_MODIFICATIONは既知のマルウェアの亜種です。
- SE_UNKNOWN_VIRUSは未知のウイルス（疑わしいオブジェクト）です。
- SE_ADWAREはアドウェアです。
- SE_DIALERはダイアラープログラムです。
- SE_JOKEはジョークプログラムです。
- SE_RISKWAREは潜在的に危険なプログラムです。

- SE_HACKTOOLはハッキングツールです。

5.脅威のリストの管理

スキャン中またはファイルシステムモニター(SpIDer Guard)によって検出された脅威のリストを管理できるように、HTTP APIには次のコマンドが用意されています。

APIコマンド	説明
threats	アクション: 検出されたすべての脅威のIDを一覧表示します。 **URI:** /api/10.2/threats/ **HTTPメソッド**: GET **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: 脅威IDの配列
site_threats	アクション: *<site ID>*というIDを持つ、Webサイトによって占有されているディレクトリで見つかったすべての脅威の脅威IDのリストを取得します。 **URI:** /api/10.2/threats/*<site ID>* **HTTPメソッド**: GET **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: 脅威IDの配列
threat_info	アクション: 脅威に関する情報を脅威のIDである*<threat ID>*で取得します。 **URI**: /api/10.2/threat_info/*<threat ID>* **HTTPメソッド**: GET **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: (*SCS-cookie*)、FileThreatオブジェクト
cure_threat	アクション: 脅威のIDである*<threat ID>*で指定された脅威の修復を試みます。 **URI**: /api/10.2/cure_threat/*<threat ID>* **HTTPメソッド**: POST **入力パラメータ**: (*SCS cookie*) **正常に実行された結果**: (*SCS cookie*)、空のオブジェクト
delete_threat	アクション: 脅威のIDである*<threat ID>*で指定された脅威を含むファイルを削除します。 **URI**: /api/10.2/delete_threat/*<threat ID>*

APIコマンド	説明
	HTTPメソッド：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト
`ignore_threat`	アクション：脅威のIDである*<threat ID>*で指定された脅威を無視します。 **URI**: `/api/10.2/ignore_threat/` *<threat ID>* **HTTP**メソッド：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト
`quarantine_threat`	アクション：脅威のIDである*<threat ID>*で指定された脅威を隔離します。 **URI**: `/api/10.2/quarantine_threat/` *<threat ID>* **HTTP**メソッド：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト

検出された脅威のリストを要求するコマンド内のURIの*<site ID>*部分は、保護されたWebサイトを含むディレクトリへのパスに置き換える必要があります。このようなディレクトリは、SpIDer Guardの設定（**LinuxSpider.Space.***<ID>***.Path**パラメータ）でも指定されます。パスは、URLエンコードの規則に従って指定する必要があります。脅威ID*<threat ID>*は、指定されたアプリケーションで見つかった脅威のID（非負の整数）です。`threat_info`、`cure_threat`、`delete_threat`、`ignore_threat`、`quarantine_threat`コマンドでは、`threats`コマンドと`site_threats`コマンドによって返される配列に含まれている脅威IDのみを使用できます。

アクション履歴を含む脅威に関するすべての情報は、`threat_info`リクエストを使用して取得できます。情報はFileThreatオブジェクトとして返されます。

1）**FileThreat**は、次のデータを含むオブジェクトです。

```
{
 "threat_id": number, //Threat identifier
 "detection_time":UNIXTime, //Time when the threat was detected
 "report":ScanReport, //Report about scanning the file
 "stat":FileStat, //Information about the file
 "origin": string, //Name of the component that detected the threat
 "origin_pid": number, //PID of the component that detected the threat
 "task_id": number, //Identifier of the scanning task
 //in the scan engine
 "history":ActionResult[] //History of actions applied to the threat (an
array)
}
```

*report*フィールドにはScanReportオブジェクトが含まれます。*stat*フィールドにはFileStatオブジェクトが含まれ、*history*フィールドにはActionResultオブジェクト（ファイルに適用されたアクションの履歴）の配列が含まれます。

2）**ScanReport** - 脅威が検出されたファイルに関する情報を含むオブジェクト。

```
{
 "object": string, //File system object that contains the threat
 "size": number, //Size (in bytes) of the file that contains the threat
 "virus":VirusInfo[], //List of details about the found
 //threats
 *"error": string, //An error message
 "heuristic_analysis": bool //Flag that shows whether heuristic
 //analysis was used
}
```

*virus*フィールドは、検出されたすべての脅威に関する情報を含むVirusInfoオブジェクトの配列です。*error*フィールドは、エラーが発生した場合にのみ表示されます。

3）**FileStat**は、ファイル統計を含むオブジェクトです。

```
{
 "dev": number, //Device containing the file
 "ino": number, //The file inode number
 *"size": number, //Size of the file
 *"uid":User, //User ID of the file's owner
 *"gid":Group, //Group ID of the owning group
 *"mode": number, //The mode of access to the file
 *"mtime":UNIXTime, //Date/time when the file was last modified
 *"ctime":UNIXTime //Date/time when the file was created
 *"rsrc_size": number, //
 *"finder_info": string, //
 *"ext_finder_info": string, //
 *"uchg": string, //
 *"volume_name": string, //Volume name
 *"volume_root": string, //Root (mount point) of the volume
 *"xattr":XAttr[] //Extended information about the file
}
```

*xattr*フィールドには、**XAttr**オブジェクトの配列が含まれています。このブジェクトには、nameおよびvalueの2つの*文字列*タイプのフィールドがあります。uidフィールドとgidフィールドにはそれぞれ**ユーザー**オブジェクトと**グループ**オブジェクトが含まれており、これらのオブジェクトにはそれぞれファイルの所有者とファイルを所有しているグループに関する情報が含まれています。これらのオブジェクトにはそれぞれ次の2つのフィールドがあります。

- uid(gid) - ユーザー（グループ）のID（数値）。
- username(groupname) - ユーザー（グループ）の名前（文字列）。

4）**ActionResult**は、ファイルに適用されたアクションとその結果に関する情報を含むオブジェクトです。

```
{
 "action": string, //The action applied
 "action_time":UNIXTime, //Date/time when the action was applied
 "result": string, //Result of applying the action
 "cure_report":ScanReport //Report about applying the action
}
```

cure_threat、delete_threat、ignore_threat、quarantine_threatコマンドは、正常に実行された場合は空のオブジェクトを返します。脅威に対して要求されたアクションがエラーのために失敗した場合（たとえば、脅威が駆除されなかった場合）、空のオブジェクトの代わりにErrorオブジェクトが返されます。

6.隔離の管理

隔離されている脅威を管理するため、HTTP APIには次のコマンドが用意されています。

APIコマンド	説明
quarantine	アクション：隔離されたオブジェクトのIDの一覧を表示します。 **URI:** /api/10.2/quarantine/ **HTTPメソッド**：GET **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、QuarantineIdオブジェクト（隔離内のオブジェクト）の配列
site_quarantine	アクション：*<site ID>*というIDを持つWebサイトのディレクトリで検出された後に隔離に移動されたすべての脅威について、隔離オブジェクトのIDの一覧を取得します。 **URI:** /api/10.2/quarantine/*<site ID>* **HTTPメソッド**：GET **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、QuarantineIdオブジェクト（隔離内のオブジェクト）の配列
qentry_info	アクション：隔離オブジェクトのIDである*<entry ID>*で指定された隔離オブジェクトに関する情報を取得します。 **URI**: /api/10.2/qentry_info/*<entry ID>* **HTTPメソッド**：GET **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、QEntryオブジェクト
cure_qentry	アクション：隔離オブジェクトのIDである*<entry ID>*で指定された隔離オブジェクトの修復を試みます。 **URI**: /api/10.2/cure_qentry/*<entry ID>* **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト

APIコマンド	説明
delete_qentry	アクション：隔離オブジェクトのIDである *<entry ID>* で指定された隔離オブジェクトを削除します。 **URI**: /api/10.2/delete_qentry/<entry ID> **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト
restore_qentry	アクション：隔離オブジェクトのIDである *<entry ID>* で指定された隔離オブジェクトを元の場所に復元します。 **URI**: /api/10.2/restore_qentry/ *<entry ID>* **HTTPメソッド**：POST **入力パラメータ**：（*SCS cookie*） **正常に実行された結果**：（*SCS cookie*）、空のオブジェクト

quarantineコマンドとsite_quarantineコマンドは、QuarantineId型のオブジェクトの配列を返します。各オブジェクトには、隔離オブジェクトのIDが含まれています。IDは、*chunk_id*と*entry_id*の2つの部分で構成されています。

1）**QuarantineId**は、隔離オブジェクトの2つの部分からなるIDの両方の部分を含むオブジェクトです。

```
{
 "chunk_id": string,
 "entry_id": string
}
```

これら2つのフィールドが一体となって隔離オブジェクトの一般的なIDを構成します。qentry_info、cure_qentry、delete_qentry、またはrestore_qentryコマンドを使用して隔離オブジェクトにアクションを適用するには、隔離オブジェクトの一般的なIDである *<entry ID>* を *<entry_id>@<chunk_id>* の形式で指定する必要があります。qentry_infoコマンドを使用すると、隔離オブジェクトのIDで指定された隔離オブジェクトに関する詳細情報を取得できます。このコマンドはQEntryタイプのオブジェクトを返します。

2）**QEntry** - 隔離オブジェクトに関する情報を含むオブジェクト

```
{
 "entry_id": string, //Parts of the identifier of
 *"chunk_id": string, //this quarantined object
 *"quarantine_dir": string, //Quarantine directory
 "restore_path": string, //path where the quarantined
 //object will be restored
 "creation_time": number, //Date/time of moving to quarantine
 //(in UNIX time)
 "report":ScanReport, //Report about scanning the object
 //(see ScanReport described above)
```

```
 "stat":FileStat, //Statistical information about the file
 //(see FileStat described above)
 *"history":QEntryOperation[], //History of operations performed on the
object
 *"who":RemoteUser, //The remote owner of the file (if
 //the file was quarantined from a file server
 //storage)
 *"detection_time": number, //Date/time of detecting the threat
 *"origin": string, //Component that detected the threat
}
```

*report*フィールドにはScanReportオブジェクトが含まれます。*stat*フィールドにはFileStatオブジェクトが含まれ、*history*フィールドには隔離オブジェクトに適用されたアクションの履歴が含まれます。各アクションエントリは、QEntryOperationオブジェクトによって記述されます。オプションの「who」フィールドには、削除されたユーザーに関する情報がRemoteUserオブジェクトの形式で含まれます。

3）**QEntryOperation**は、隔離オブジェクトに適用された操作に関するデータを含むオブジェクトです

```
{
 "action": string, //Operation performed on the object
 //(see the possible values below)
 "action_time": number, //Date/time when the operation was performed
(UNIX Time)
 "result": string, //Error when trying to perform the operation (a code
 //EC_XXX)
 *"restore_path": string, //path for restoring the quarantined object
 //(if action = "QENTRY_ACTION_RESTORE")
 *"rescan_report":ScanReport //Report about rescanning (if
 //action = "QENTRY_ACTION_RESCAN")
}
```

*action*フィールドには、以下の値を指定できます。

- QENTRY_ACTION_DELETEは、隔離オブジェクトの削除を試みます。
- QENTRY_ACTION_RESTOREは、隔離オブジェクトの復元を試みます。
- QENTRY_ACTION_RESCANは、隔離オブジェクトの再スキャンを試みます。
- QENTRY_ACTION_CUREは、隔離オブジェクトの修復を試みます。

4）**RemoteUser**は、ファイルを所有するリモートユーザーに関する情報を含むオブジェクトです（ファイルがファイルサーバーストレージから隔離に再配置された場合）。

```
{
 *"ip": string, //IP-address of the user
 *"user": string, //User name
 *"domain": string //Domain of the user
}
```

cure_qentry、delete_qentry、restore_qentryコマンドの実行が成功すると、空のオブジェクトが返されます。隔離オブジェクトに対して要求された操作がエラーで終了した場合（たとえば、ファイルを復元できなかった場合）、空のオブジェクトの代わりにErrorオブジェクトが返されます。

7.HTTP APIの使用例

HTTP APIの動作をテストするには、**curl**ユーティリティを使用します。API呼び出しの一般的なフォーマットは以下のとおりです。

```
$ curl https://<HTTPD.AdminListen>/<HTTP API URI> -k -X <HTTP method name>
 [-H 'Content-Type: application/json' --data-binary '@<file of the JSON object>']
 [-c <cookie file> [-b <cookie file>]] [> <file of the result>]
```

ここで、-kコマンドラインパラメータは、使用されるSSL証明書が有効であるかどうかを**curl**がチェックしないことを許可するために使用され、-Xパラメータは、使用するHTTPメソッド(GETまたはPOST)を指定するために使用されます。JSONオブジェクトがこのHTTPリクエストの本文で渡される場合は、-Hパラメータを使用してリクエストのヘッダーセクションにContent-Type: application/jsonヘッダーフィールドを追加できます。また、--data-binaryパラメータを使用すると、JSONオブジェクト自体をリクエストの本文に追加し、テキストファイルからJSONオブジェクトを取得できます。SCSを使用して認証を受ける場合は、送受信される*SCS cookie*を格納するファイルをそれぞれ-bパラメータと-cパラメータで指定する必要があります。このユーティリティとその引数に関する詳細な説明を表示するには、**curl** --helpコマンドと**man** curlコマンドを使用してください。

1. ユーザー名とパスワード(SCS用)を指定して、**クライアントを認証および承認する**。

   JSON形式のAuthOptionsオブジェクトがあらかじめuser.jsonというファイルに書き込まれている必要があります。例:

   {"user":"*<ユーザー名>*","password":"*<パスフレーズ>*"}。

   ```
   $ curl https://127.0.0.1:4443/api/10.2/login -k -X POST -H 'Content-Type:
   application/json' --data-binary '@user.json' -c cookie.file
   ```

   レスポンス:

   ```
   HTTP/1.0 200 OK
   Content-Type: application/json
   Content-Length:2
   Set-
   Cookie:DWToken=6QXy4wn_JGov9A1GohWP_kvMK3dN6ccKegjNgKcmHpb_AqSrHg9cNX_yFJ
   hxPDgr|MTQ2Mjg3Mzg4NQ==|cWd4Ow==|GywBUVOhU4w2LF_BKT5frg==|
   kR_rip5nrpxWjJ2dfZ7Xfmvi3rE=; Secure; HttpOnly; Max-Age:900; Path=/
   Pragma: no-cache

   {}
   ```

   Set-Cookieヘッダーフィールドには、HTTP APIに次のコマンドを送信するために使用する必要がある*SCS-cookie*が含まれています。認証と承認が成功した場合、レスポンスの本文には空のオブジェクトが含まれています。ユーザーが承認されなかった場合は、次のようなErrorオブジェクトが返されます。

   ```
   HTTP/1.0 403 Forbidden
   Content-Type: application/json
   Content-Length:35
   Pragma: no-cache

   {"error":{"code":"EC_AUTH_FAILED"}}
   ```

2. Webサイトのディレクトリで検出された**脅威のリストを取得する**(ここではサイトのルートディレクトリとして/sites/site1ディレクトリが指定されています):

```
$ curl https://127.0.0.1:4443/api/10.2/threats/%2Fsites%2Fsite1 -k -X GET
-c cookie.file -b cookie.file
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:80
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

["1", "2", "3", "4"]
```

3. IDが*1*である脅威に関する情報を取得する:

```
$ curl https://127.0.0.1:4443/api/10.2/threat_info/1 -k -X GET -c
cookie.file -b cookie.file
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:574
Set-Cookie:DWToken=<...>;
 Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"threat_id":1,"detection_time":1462881660,
"report":{"object":"/sites/site1/eicar.com.txt","size":68,"packer":[],
"virus":[{"type":"SE_KNOWN_VIRUS","name":"EICAR Test File (NOT a
Virus!)"}],
"heuristic_analysis":true,"core_fingerprint":"0D2DD5A869DAB7AE354153A4D5F
70F45",
"item":[],"log":[],"user_time":0,"system_time":0},"stat":
{"dev":2049,"ino":898,
"size":68,"uid":{"uid":1000,"username":"user"},"gid":
{"gid":1000,"groupname":"user"},
"mode":33204,"mtime":1441028214,"ctime":1460738554,"xattr":[]},
"origin":"APP_COMMAND_LINE_TOOL","origin_pid":2726,"task_id":1,"history":
[]}
```

4. IDが*1*である脅威を隔離に移動する:

```
$ curl -v -c cookie.jar -b cookie.jar -k -X POST -H 'Content-
Type:application/json'
https://127.0.0.1:4443/api/10.2/quarantine_threat/1
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:2
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache
```

```
{}
```

5. 指定されたWebサイトの**隔離のコンテンツを表示する**:

```
$ curl -v -k -X GET -c cookie.jar -b cookie.jar
https://127.0.0.1:4443/api/10.2/quarantine/%2Fsites%2Fsite1
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:76
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

[{"entry_id":"3070d3ce-7b6e-4143-9d9f-
89ba3473a781","chunk_id":"801:2108d"}]
```

6. **隔離(分離)オブジェクトに関する情報を表示する**:

```
$ curl -v -k -X GET -c cookie.jar -b cookie.jar
https://127.0.0.1:4443/api/10.2/qentry_info/3070d3ce-7b6e-4143-9d9f-
89ba3473a781@801:2108d
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:781
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"entry_id":"3070d3ce-7b6e-4143-9d9f-
89ba3473a781","chunk_id":"3830313A3231303864",
"quarantine_dir":"2F686F6D652F757365722F2E636F6D2E64727765622E71756172616
E74696E65",
"restore_path":"2E2E2F7473742F65696361722E636F6D2E747874","creation_time"
:1462888884,
"report":{"object":"/home/user/tst/eicar.com.txt","size":68,"packer":[],
"virus":[{"type":"SE_KNOWN_VIRUS","name":"EICAR Test File (NOT a
Virus!)"}],
"heuristic_analysis":true,"core_fingerprint":"467CD4C6D423C55448B71CD5B81
52776",
"item":[],"log":[],"user_time":0,"system_time":0},"stat":
{"dev":2049,"ino":898,
"size":68,"uid":{"uid":1000,"username":"user"},"gid":
{"gid":1000,"groupname":"user"},
"mode":33204,"mtime":1441028214,"ctime":1462888421,"xattr":[]},"history":
[],
"detection_time":1462888667,"origin":"APP_COMMAND_LINE_TOOL"}
```

7. **設定の変更**: SpIDer Guardをオフにする。

JSON形式のLexMapオブジェクトがあらかじめlexmap_ls_off.jsonというファイルに書き込まれている必要があります。

```
{"option":[{"key":"LinuxSpider","map":{"option":
[{"key":"Start","value":{"item":["no"]}}]}}]}
```

```
$ curl -v -k -c cookie.jar -b cookie.jar -X POST -H 'Content-Type:
application/json' --data-binary '@lexmap_ls_off.json'
https://127.0.0.1:4443/api/10.2/set_lexmap
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:58
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"item":[{"option":"LinuxSpider.Start","result":"EC_OK"}]}
```

8. **設定の変更**: SpIDer Guardをオンにする。

   JSON形式のLexMapオブジェクトがあらかじめlexmap_ls_on.jsonというファイルに書き込まれている必要があります。

   ```
   {"option":[{"key":"LinuxSpider","map":{"option":
   [{"key":"Start","value":{"item":["yes"]}}]}}]}
   ```

   ```
   $ curl -v -k -c cookie.jar -b cookie.jar -X POST -H 'Content-Type:
   application/json' --data-binary '@lexmap_ls_on.json'
   https://127.0.0.1:4443/api/10.2/set_lexmap
   ```

   レスポンス:

   ```
   HTTP/1.0 200 OK
   Content-Type: application/json
   Content-Length:58
   Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
   Pragma: no-cache

   {"item":[{"option":"LinuxSpider.Start","result":"EC_OK"}]}
   ```

9. **設定の変更**: ホストされているWebサイトをSpIDer Guardの保護下に置く。

   このアクションは2つのステップで実行されます。まず、個別のサイト（SpIDer Guardでは「保護スペース」）のセクションを追加する必要があります。次に、サイトを保護するためのパラメータを指定する必要があります（少なくとも、サイトのファイルを格納するディレクトリへのパスを指定する必要があります）。

   1) **Webサイトを追加する。**JSON形式のLexMapオブジェクトがあらかじめlexmap_site.jsonというファイルに書き込まれている必要があります。

      ```
      {"option":[{"key":"LinuxSpider","map":{"option":
      [{"key":"Space","map":{"option":[{"key":"<SITE_ID>","map":
      {"option":[]}}]}}]}}]},
      ```

      ここで、*<SITE_ID>*は目的のWebサイトのIDです。

      ```
      $ curl -v -k -c cookie.jar -b cookie.jar -X POST -H 'Content-Type:
      application/json' --data-binary '@lexmap_site.json'
      https://127.0.0.1:4443/api/10.2/set_lexmap
      ```

      レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:11
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"item":[]}
```

設定ファイルに新しく作成されたセクションにはまだパラメータが設定されていないため、このリクエストに対するレスポンスでは空のオブジェクトが返されます。

2) **Webサイトのディレクトリへのパスを設定する。**JSON形式のLexMapオブジェクトがあらかじめlexmap_site_path.jsonというファイルに書き込まれている必要があります。

```
{"option":[{"key":"LinuxSpider","map":{"option":
[{"key":"Space","map":{"option":[{"key":"<SITE_ID>","map":
{"option":[{"key":"Path","value":{"item":["<PATH>"]}}]}}]}}]}}]}},
```

ここで、*<SITE_ID>*は保護するWebサイトのID、*<PATH>*はこのサイトを格納しているディレクトリへのフルパスです。

```
$ curl -v -k -c cookie.jar -b cookie.jar -X POST -H 'Content-Type:
application/json' --data-binary '@lexmap_site_path.json'
https://127.0.0.1:4443/api/10.2/set_lexmap
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:65
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"item":[{"option":"LinuxSpider.Space.s2.Path","result":"EC_OK"}]}
```

この例では、s2は、SpIDer Guardによって保護されるディレクトリへのパスを指定したWebサイトのIDです。

10. **設定の変更**：Webサイトの保護をオフにする。

JSON形式のLexMapオブジェクトがあらかじめlexmap_site_off.jsonというファイルに書き込まれている必要があります。

```
{"option":[{"key":"LinuxSpider","map":{"option":
[{"key":"Space","map":{"option":[{"key":"<SITE_ID>","map":{"option":
[{"key":"Enable","value":{"item":["No"]}}]}}]}}]}}]}},
```

ここで、*<SITE_ID>*は目的のWebサイトのIDです。

```
$ curl -v -k -c cookie.jar -b cookie.jar -X POST -H 'Content-Type:
application/json' --data-binary '@lexmap_site_off.json'
https://127.0.0.1:4443/api/10.2/set_lexmap
```

レスポンス:

```
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length:67
```

```
Set-Cookie:DWToken=<...>; Secure; HttpOnly; Max-Age:900; Path=/
Pragma: no-cache

{"item":[{"option":"LinuxSpider.Space.s2.Enable","result":"EC_OK"}]}
```

この例では、*s2*は保護が無効になっている目的のWebサイトのIDです。

Dr.Web SNMPD

Dr.Web SNMPDは、SNMPプロトコルを実行するモニタリングシステムにDr.Web for UNIX Internet Gatewaysを統合するよう設計されたSNMPエージェントです。この統合により、Dr.Web for UNIX Internet Gatewaysコンポーネントのステータスを追跡したり、脅威の検出と無効化に関する統計を収集したりできます。エージェントは、モニタリングシステムまたはSNMPマネージャーに以下の情報を提供するサポートをします。

- 任意のDr.Web for UNIX Internet Gatewaysコンポーネントのステータス。
- 検出されたさまざまなタイプの脅威の数（Dr.Web分類に対応）。

さらに、エージェントは、脅威を検出したときと検出された脅威の駆除が失敗したときにSNMPトラップを送信します。エージェントはSNMPプロトコルバージョン2cおよび3をサポートします。

エージェントが送信できる情報の説明は、Doctor Webによって作成されたMIB（*管理情報ベース*）の特別なセクションに格納されています。UNIX系オペレーティングシステム用にDr.Webが定義したMIBセクションには、以下の情報が指定されています。

1. 脅威の検出と駆除、Dr.Web for UNIX Internet Gatewaysコンポーネントに関連するエラーに関するSNMPトラップ通知のフォーマット。
2. Dr.Web for UNIX Internet Gateways操作の統計。
3. Dr.Web for UNIX Internet Gatewaysコンポーネントのステータス。

SNMPプロトコルを介して取得できる情報の詳細については、対応するセクションを参照してください。

動作原理

このセクションの内容：

- 機能
- システムSNMPエージェントとの統合

概要

デフォルトでは、コンポーネントはDr.Web for UNIX Internet Gatewaysの起動時に自動的に実行されます。実行されると、コンポーネントはMIB Dr.Webで記述されている構造に従ってデータを構造化し、外部SNMPマネージャーからのデータ受信要求を待機します。コンポーネントは、Dr.Web ConfigD設定デーモンから、Dr.Web for UNIX Internet Gatewaysコンポーネントのステータスに関する情報と検出された脅威に関する通知を直接受信します。

脅威は、Dr.Web for UNIX Internet Gatewaysコンポーネントによるスキャン中にスキャンエンジンによって検出されます。何らかの脅威が検出されると、（この脅威の種類に）該当するカウントが1つ増え、通知を受信できるすべてのSNMPマネージャーは、検出された脅威について通知するSNMPトラップを受け取ります。

カウンターの収集値（たとえば、検出された脅威のカウンター）は、Dr.Web SNMPDの起動の合間は保存されません。したがって、Dr.Web SNMPDが何らかの理由（Dr.Web for UNIX Internet Gatewaysの一般的な再起動を含む）で再起動されると、収集されたカウンターの値は0にリセットされます。

システムSNMPエージェントとの統合

メインシステムのSNMPエージェント**snmpd**（**net-snmp**）がすでにサーバー上で動作している場合にDr.Web SNMPエージェントが正しく動作するようにするには、Dr.Web MIBブランチを介した**snmpd**からDr.Web SNMPDへのSNMPリクエスト送信を設定します。そのためには、次の行を追加して**snmpd**設定ファイル（**GNU/Linux**の場合は、通常`/etc/snmp/snmpd.conf`）を編集します。

```
proxy -v <version> -c <community> <address>:<port> <MIB branch>
```

- *<version>* - 使用中のSNMPバージョン（`2c`、`3`）。
- *<community>* - Dr.Web SNMPDによって使用される「コミュニティストリング」。
- *<address>:<port>* - Dr.Web SNMPDによって待ち受け（リッスン）されるネットワークソケット。
- *<MIB branch>* - Dr.Webが使用する変数とSNMP*トラップ*の説明を含むMIBブランチのOID（OIDは`.1.3.6.1.4.1.29690`）。

Dr.Web SNMPエージェントをデフォルト設定で使用している場合、追加する行は以下のようになります。

```
proxy -v 2c -c public localhost:50000 .1.3.6.1.4.1.29690
```

この場合、ポート161はシステムの標準**snmpd**によって使用されるため、ListenAddressパラメータにDr.Web SNMPDに別のポートを指定する必要があります（この例では、`50000`）。

コマンドライン引数

オペレーティングシステムのコマンドラインからDr.Web SNMPDを起動するには、次のコマンドを使用します。

```
$ <opt_dir>/bin/drweb-snmpd [<parameters>]
```

Dr.Web SNMPDは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-snmpd --help
```

このコマンドはDr.Web SNMPDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じてDr.Web ConfigD設定デーモンによって自動的に起動されます（原則として、OSの起動時）。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-snmpd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[SNMPD]`セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、`[Root]`セクションの**DefaultLogLevel**パラメータの値が使用されます。

	デフォルト値：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-snmpd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-snmpd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-snmpd
Start *{Boolean}*	コンポーネントはDr.Web ConfigD設定デーモンによって起動される必要があります。 このパラメータにYes値を指定すると、設定デーモンはただちにコンポーネントを開始するように指示されます。また、No値を指定すると、設定デーモンはただちにコンポーネントを終了するように指示されます。 **デフォルト値**：No
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。 **RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
ListenAddress *{address}*	クライアント接続（SNMPマネージャー）を待機するDr.Web SNMPDが待ち受け（リッスン）するアドレス（IPアドレスとポート）。 **snmpd**とのインタラクションには、*標準ポート（161）とは異なるポートの指定と、***snmpd***にプロキシ用の設定を行う必要があります。* **デフォルト値**：127.0.0.1:161
SnmpVersion *{V2c / V3}*	使用されるSNMPプロトコルのバージョン（*SNMPv2c*または*SNMPv3*）。 **デフォルト値**：V2c
V3EngineId *{string}*	*SNMPv3のエンジンID*の識別子（文字列）（RFC 3411に準拠）。 **デフォルト値**：800073FA044452574542
TrapReceiver	Dr.Web for UNIX Internet Gatewaysコンポーネントが脅威を検出した後にDr.Web SNMPDによって*SNMPトラップ*が送信さ

{address list}	れるアドレスのリスト（IPアドレスとポート）。 *リストをパラメータ値として指定できます。リストの値は、コンマ（引用符内の各値）で区切る必要があります。パラメータはセクションで複数回指定できます（この場合、そのすべての値が1つのリストにまとめられます）。* **例**：ソケット192.168.0.1:1234および10.20.30.45:5678をリストに追加します。 1. 設定ファイルに値を追加します。 • 1つの文字列に2つの値 セクション[SNMPD] TrapReceiver = "192.168.0.1:1234", "10.20.30.45:5678" • 2つの文字列（文字列ごとに1つの値） [SNMPD] TrapReceiver = 192.168.0.1:1234 TrapReceiver = 10.20.30.45:5678 2. コマンド**drweb-ctl** cfsetを使用して値を追加します。 # **drweb-ctl** cfset SNMPD.TrapReceiver -a 192.168.0.1:1234 # **drweb-ctl** cfset SNMPD.TrapReceiver -a 10.20.30.45:5678 **デフォルト値**：*（未設定）*
V2cCommunity *{string}*	Dr.Web MIB変数が読み取りアクセスされたときに、SNMPマネージャー（*SNMPv2c*プロトコル）を認証する文字列「SNMP read community」。 **SnmpVersion** = V2c*の場合、このパラメータが使用されます。* **デフォルト値**：public
V3UserName *{string}*	Dr.Web MIB変数が読み取りアクセスされたときに、SNMPマネージャー（*SNMPv3*プロトコル）を認証するユーザー名。 **SnmpVersion** = V3*の場合、このパラメータが使用されます。* **デフォルト値**：noAuthUser
V3Auth *{SHA(<pwd>) | MD5(<pwd>) | None}*	Dr.Web MIB変数が読み取りアクセスされたときに、SNMPマネージャー（*SNMPv2c*プロトコル）を認証する方法。 **使用可能な値**： • SHA（*<PWD>*）-パスワードのSHAハッシュが使用されます（*<PWD>*文字列）。

	• MD5 (*<PWD>*) -パスワードのMD5ハッシュが使用されます (*<PWD>*文字列)。 • None-認証は無効になります。 ここで、*<PWD>*はプレーンテキストのパスワードです。 コマンドラインからパラメータ値を指定する場合、シェルによってはスラッシュ記号 \を使用した角括弧のエスケープを必要とする場合があります。 **例**: 1. 設定ファイル内のパラメータ値: `V3Auth = MD5(123456)` 2. コマンド**drweb-ctl** cfsetを使用して、コマンドラインから同じパラメータ値を指定する場合: `drweb-ctl cfset SNMPD.V3Auth MD5\(123456\)` **SnmpVersion** = V3*の場合、このパラメータが使用されます。* **デフォルト値**: なし
V3Privacy *{DES(<secret>) | AES128(<secret>) | None}*	SNMPメッセージを暗号化する方法 (*SNMPv3*プロトコル)。 **使用可能な値**: • DES (*<secret>*) -DES暗号化アルゴリズムが使用されます。 • AES128 (*<secret>*) - AES128暗号化アルゴリズムが使用されます。 • None-SNMPメッセージは暗号化されません。 ここで、*<secret>*は、マネージャーとエージェントが共有するシークレットキーです (*プレーンテキスト*)。 コマンドラインからパラメータ値を指定する場合、シェルによってはスラッシュ記号 \を使用した角括弧のエスケープを必要とする場合があります。 **例**: 1. 設定ファイル内のパラメータ値: `V3Privacy = AES128(supersecret)` 2. コマンド**drweb-ctl** cfsetを使用して、コマンドラインから同じパラメータ値を指定する場合: `drweb-ctl cfset SNMPD.V3Privacy AES128\(supersecret\)` **SnmpVersion** = V3*の場合、このパラメータが使用されます。* **デフォルト値**: なし

SNMPモニタリングシステムとの統合

Dr.Web SNMPエージェントは、SNMPプロトコル、バージョン*2c*または*3*を使用するモニタリングシステムのデータプロバイダーとして機能します。制御に使用できるデータのリストとデータ構造は、Dr.Web for UNIX Internet Gatewaysに付属のDr.Web MIB記述ファイル`DRWEB-SNMPD-MIB.txt`に記述されています。このファイルは、ディレクトリ*<opt_dir>*`/share/drweb-snmpd/mibs`にあります。

簡単に設定できるように、このコンポーネントには一般的なモニタリングシステム用の設定テンプレートが付属しています。

- Munin
- Nagios
- Zabbix

モニタリングシステム用のカスタマイズテンプレートは、*<opt_dir>*`/share/drweb-snmpd/connectors`ディレクトリ にあります。

Muninモニタリングシステムとの統合

Muninモニタリングシステムには、モニタリング対象ホストにローカルに存在するクライアント**munin-node**から統計を収集する中央サーバー(マスター)**munin**が含まれています。サーバーの要求に応じて、各モニタリングクライアントは、サーバーに転送されたデータを提供する*プラグイン*(*プラグイン*)を起動することによって、モニタリング対象ホストの動作に関するデータを収集します。

Dr.Web SNMPDと**Munin**モニタリングシステム間の接続を可能にするために、すぐに利用可能な**munin-node**用プラグインが提供されています。プラグインは、*<opt_dir>*`/share/drweb-snmpd/connectors/munin/plugins`ディレクトリにあります。このプラグインは、下の2つのグラフの作成に必要なデータを収集します。

- 検出された脅威の数
- ファイルのスキャン統計情報
- メールメッセージのスキャン統計情報

これらのプラグインは、SNMPプロトコルバージョン1、2c、3をサポートしています。これらのテンプレートプラグインに基づいて他のプラグインを作成すれば、Dr.Web SNMPD経由でDr.Web for UNIX Internet Gatewaysコンポーネントのステータスをポーリングすることができます。

<opt_dir>`/share/drweb-snmpd/connectors/munin`ディレクトリには、下のファイルがあります。

ファイル	説明
`plugins/snmp__drweb_malware`	ホスト上の Dr.Web for UNIX Internet Gatewaysによって検出された脅威の数を収集するために、SNMP経由でDr.Web SNMPDをポーリングするための**munin-node**プラグイン。
`plugins/snmp__drweb_filecheck`	ホスト上のDr.Web for UNIX Internet Gatewaysによってスキャンされたファイルの統計を収集するために、SNMP経由でDr.Web SNMPDをポーリングするための**munin-node**プラグイン。

ファイル	説明
plugins/snmp__drweb_maild_multi	ホスト上のDr.Web for UNIX Internet Gatewaysによってスキャンされたメールメッセージの統計を収集するために、SNMP経由でのDr.Web SNMPDのポーリングに使用される**munin-node**プラグイン。 このプラグインは、**Munin**バージョン1.4以上で利用可能な機能*multigraph*を使用します。
plugin-conf.d/drweb.cfg	Dr.Webプラグインの**munin-node**設定の例（環境変数用）。

Muninにホストを接続する

この説明では、**Munin**モニタリングシステムがモニタリングサーバーにすでにデプロイされており、モニタリング対象ホストにはDr.Web SNMPDがインストール済みで**munin-node**と共に機能していると想定しています（コンポーネントは**snmpd**と共にプロキシモードで機能することができます）。

1. モニタリング対象ホストの設定
   - snmp__drweb_*ファイルを、プラグインライブラリ**munin-node**があるディレクトリにコピーします（ディレクトリはOSによって異なります）。たとえば、**Debian/Ubuntu**オペレーティングシステムであれば、パスは/usr/share/munin/pluginsとなります。
   - 提供されているDr.Webプラグインを接続して、**munin-node**を設定します。これを行うには、**munin-node**と一緒に配布される**munin-node-configure**ユーティリティを使用します。

     たとえば、次のようなコマンドです。

     ```
     $ munin-node-configure --shell --snmp localhost
     ```

     端末画面に、プラグインに必要なシンボリックリンクを作成するためのコマンドのリストを表示します。コピーして、コマンドラインで実行します。指定されたコマンドでは、次のことを前提としています。

     1) **munin-node**は、Dr.Web SNMPDがインストールされているのと同じホストにインストールされる。インストール先が異なる場合は、localhost値ではなく、モニタリング対象ホストの適切なFQDNまたはIPアドレスを指定してください。
     2) Dr.Web SNMPDはSNMPバージョン2cを使用する。他のバージョンの場合は、**munin-node-configure**コマンドで適切なSNMPバージョンを指定してください。このコマンドには、プラグインを柔軟に設定するための引数がいくつかあります。たとえば、SNMPプロトコルのバージョン、モニタリング対象ホストでSNMPエージェントが待機しているポート、*コミュニティストリング*の実際の値などを指定できます。必要に応じて、**munin-node-configure**コマンドのマニュアルを参照してください。
   - 必要であれば、**munin-node**用にインストールしたDr.Webプラグインを実行する環境のパラメータ値を定義（または再定義）します。環境パラメータとして、値*コミュニティストリング*が使用されます。これは、SNMPエージェントなどが使用するポートです。これらのパラメータは、ファイル/etc/munin/plugin-conf.d/drwebに定義する必要があります（必要に応じて作成します）。このファイルの例として、提供されているファイルdrweb.cfgを使用します。
   - **munin-node**設定ファイル（munin-node.conf）で、モニタリング対象のパラメータの値を受け取るために**munin**サーバー（マスター）を**munin-node**に接続できるホストのすべてのIPアドレスを含めるための正規表現を指定します。たとえば、次のようになります。

```
allow ^10\.20\.30\.40$
```

この場合、IPアドレス10.20.30.40のみがホストパラメータを受信できます。

- たとえば、次のコマンドを使用して**munin-node**を再起動します。

```
# service munin-node restart
```

2. **Munin**サーバー(マスター)の設定

   モニタリング対象ホストのアドレスと識別子を**Munin**設定ファイルmunin.confに追加します。このファイルは、デフォルトでは/etcディレクトリにあります(**Debian/Ubuntu**オペレーティングシステムでは/etc/munin/munin.confになります)。

```
[<ID>; <hostname>. <domain>]
address <host IP address>
use_node_name yes
```

   ここで、*<ID>*は表示されるホストの識別子、*<hostname>*はホストの名前、*<domain>*はドメインの名前、*<host IP address>*はホストのIPアドレスです。

Muninモニタリングシステムの設定に関する公式マニュアルについてはhttp://munin.readthedocs.ioを参照してください。

Zabbixモニタリングシステムとの統合

Dr.Web SNMPDと**Zabbix**モニタリングシステム間の接続を確立するために必要なファイルテンプレートは、*<opt_dir>*/share/drweb-snmpd/connectors/zabbixディレクトリにあります。

ファイル	説明
zbx_drweb.xml	Dr.Web for UNIX Internet Gatewaysがインストールされているモニタリング対象ホストを説明するためのテンプレート。
snmptt.drweb.zabbix.conf	*SNMPトラップ*ハンドラーである**snmptt**ユーティリティの設定

モニタリング対象ホストの機能を説明するためのテンプレート。

- カウンター(**Zabbix**の用語では「*アイテム*」)の説明。デフォルトでは、テンプレートはSNMP v2で使用されるように設定されています。
- 既存のグラフのセット:スキャンされたファイルの数と検出された脅威のタイプ別の分布。

Zabbixにホストを接続する

この説明では、**Zabbix**モニタリングシステムがモニタリングサーバーにすでにデプロイされており、モニタリング対象ホストにはDr.Web SNMPDがインストール済みで機能していると想定しています（コンポーネントは**snmpd**と共にプロキシモードで機能することができます）。さらに、モニタリング対象ホストから*SNMP*トラップ（保護対象サーバーでDr.Web for UNIX Internet Gatewaysによって検出された脅威に関する通知を含む）を受信する場合は、モニタリングサーバーに`net-snmp`パッケージをインストールします（標準ツールの**snmptt**および**snmptrapd**が使用されます）。

1. **Zabbix** Webインターフェースの**Configuration** -> **Templates**タブで、*<opt_dir>*`/share/drweb-snmpd/connectors/zabbix/zbx_drweb.xml`ファイルからモニタリング対象ホストのテンプレートをインポートします。
2. モニタリング対象ホストを適切なリストに追加します（**Hosts** -> **Create host**）。ホストの適切なパラメータとSNMPインターフェースの設定を指定します（ホストの**drweb-snmpd**と**snmpd**の設定と一致する必要があります）。
   - **Host**タブ：

     Host name：*drweb-host*

     Visible name：*DRWEB_HOST*

     Groups：*Linux servers*を選択します

     Snmp interfaces：**Add**をクリックして、Dr.Web SNMPDによって使用されるIPアドレスとポート（Dr.Web SNMPDはローカルホストで動作すると見なされるため、デフォルトではアドレス*127.0.0.1*とポート*161*が指定されています）を指定します。
   - **Templates**タブ：

     Addを押し、*DRWEB*を確認し、**Select**を押します。
   - **Macros**タブ：

     Macro：*{$SNMP_COMMUNITY}*

     Value：SNMP V2cに「read community」を指定します（デフォルトでは、*public*）。

     Saveをクリックします。

     注意：*{$SNMP_COMMUNITY}*マクロは、ホストテンプレートで直接指定できます。

   

   デフォルトでは、インポートされた*DRWEB*テンプレートはSNMP v2用に設定されています。他のバージョンのSNMPを使用する場合は、該当するページでテンプレートに必要な編集を行います。

3. テンプレートがモニタリング対象ホストにバインドされた後、SNMP設定が正しく指定されていれば、**Zabbix**モニタリングシステムはテンプレートのカウンター（*アイテム*）のデータ収集を開始します。収集されたデータは、**Monitoring** -> **Latest Data**と**Monitoring** -> **Graphs**に表示されます。
4. Dr.Web SNMPDから*SNMPトラップ*通知を収集するために特別な*アイテムdrweb-traps*が使用されます。受信した*SNMPトラップ*通知のログは、**Monitoring**→**Latest Data**→**drweb-traps** –> **history**ページで利用できます。**Zabbix**は、通知を収集するために、`net-snmp`パッケージの標準ツール**snmptt**と**snmptrapd**を使います。Dr.Web SNMPDから*SNMPトラップ*通知を受信するためのツールの設定方法については、以下を参照してください。
5. 必要に応じて、Dr.Web SNMPDからの*SNMPトラップ*通知の受信時に状態を変更するトリガーを設定できます。状態の変更は、適切な通知を生成するためのイベントソースとして使用できます。下の例は、トリガーの設定式を示します。この式は**trigger expression**フィールドで指定されます。

- **Zabbix**バージョン*2.x*の場合：

```
({TRIGGER.VALUE}=0 &
{DRWEB:snmptrap[.*\.1\.3\.6\.1\.4\.1\.29690\..*].nodata(60)}=1 )|
({TRIGGER.VALUE}=1 &
{DRWEB:snmptrap[.*\.1\.3\.6\.1\.4\.1\.29690\..*].nodata(60)}=0)
```

- **Zabbix**バージョン*3.x*の場合：

```
({TRIGGER.VALUE}=0 and {drweb-host:snmptrap[".29690."].nodata(60)}=1 ) or
({TRIGGER.VALUE}=1 and {drweb-host:snmptrap[".29690."].nodata(60)}=0 )
```

Dr.Web SNMPDからの*SNMPトラップ*のログが1分以内に更新された場合、イベントがトリガー（値を1に設定）されます。ログが次の1分以内に更新されなかった場合、トリガーの値は再び0に設定されます。

Severityでは、このトリガーの通知タイプを*Not classified*とは異なるものにすることをお勧めします（例：*Warning*）。

ZabbixのSNMPトラップ通知の受信を設定する

1. モニタリング対象ホストのDr.Web SNMPD設定（`TrapReceiver`パラメータ）で、**Zabbix**が動作しているホストで**snmptrapd**が待ち受け（リッスン）するアドレスを指定する必要があります。次に例を示します。

```
SNMPD.TrapReceiver = 10.20.30.40:162
```

2. **snmptrapd**の設定ファイル（`snmptrapd.conf`）に、同じアドレスと、受信した*SNMPトラップ*通知を処理するアプリケーションを指定します（この例では**snmptthandler**、**snmptt**コンポーネント）。

```
snmpTrapdAddr 10.20.30.40:162
traphandle default /usr/sbin/snmptthandler
```

Dr.Web SNMPDによって送信された*SNMPトラップ*を**snmptt**が不明なものとして破棄しないように、ファイルに次の文字列を追加します。

```
outputOption n
```

3. **snmptthandler**コンポーネントは、**Zabbix**のホストテンプレートに設定されている正規表現（*アイテム drweb-traps*エレメント）に対応する指定された形式に従って、受信した*SNMPトラップ*通知をディスクのファイルに保存します。保存された通知の*SNMPトラップ*フォーマットは、*<opt_dir>*`/share/drweb-snmpd/connectors/zabbix/snmptt.drweb.zabbix.conf`ファイルで指定します。ファイルは、`/etc/snmp`にコピーする必要があります。
4. さらに、フォーマットファイルへのパスを`snmptt.ini`に指定する必要があります。

```
[TrapFiles]
# snmptt.conf ファイルのリスト（これはsnmptrapd.conf ファイルではありません）。
# 完全なパスとファイル名。例：'/etc/snmp/snmptt.conf'
snmptt_conf_files = <<END
/etc/snmp/snmptt.conf
/etc/snmp/snmptt.drweb.zabbix.conf
END
```

その後、デーモンモードで起動している場合は**snmptt**を再起動します。

5. **Zabbix**サーバーの設定ファイル（`zabbix-server.conf`）で、次の設定を指定します（または、すでに指定されているかどうかを確認します）。

```
SNMPTrapperFile=/var/log/snmptt/snmptt.log
StartSNMPTrapper=1
```

ここで、`/var/log/snmptt/snmptt.log`は、受信したSNMPトラップに関する情報を登録するために**snmptt**が使用するログファイルです。

Zabbixの公式マニュアルについては、https://www.zabbix.com/documentation/を参照してください。

Nagiosモニタリングシステムとの統合

Dr.Web SNMPD と**Nagios**モニタリングシステム間の接続を確立するために必要なファイルと**Nagios**設定例は、*<opt_dir>*`/share/drweb-snmpd/connectors/nagios`ディレクトリにあります。

ファイル	説明
`nagiosgraph/rrdopts.conf-sample`	RRD設定ファイルの例
`objects/drweb.cfg`	*drweb*オブジェクトを記述する設定ファイル
`objects/nagiosgraph.cfg`	**Nagios**が使用する**Nagiosgraph**によって使用されるグラフ描画のためのコンポーネントの設定ファイル
`plugins/check_drweb`	Dr.Web for UNIX Internet Gatewaysがインストールされているホストからデータを収集するためのスクリプト
`plugins/eventhandlers/submit_check_result`	*SNMPトラップ*通知を処理するためのスクリプト
`snmp/snmptt.drweb.nagios.conf`	*SNMPトラップ*ハンドラーである**snmptt**ユーティリティの設定

Nagiosにホストを接続する

この説明では、Webサーバーとグラフィックツール**Nagiosgraph**の設定を含む**Nagios**モニタリングシステムがモニタリングサーバーにすでにデプロイされており、モニタリング対象ホストにはDr.Web SNMPDがインストール済みで機能していると想定しています（コンポーネントは**snmpd**と共にプロキシモードで機能することができます）。さらに、モニタリング対象ホストから*SNMPトラップ*（保護対象サーバーでDr.Web for UNIX Internet Gatewaysによって検出された脅威に関する通知を含む）を受信する場合は、モニタリングサーバーに`net-snmp`パッケージをインストールします（標準ツールの**snmptt**および**snmptrapd**が使用されます）。

現在のマニュアルでは、次のようなパスの規則が使用されています（実際のパスはオペレーティングシステムと**Nagios**のインストールによって異なります）。

- *<NAGIOS_PLUGINS_DIR>* - **Nagios**プラグインを含むディレクトリ（例：`/usr/lib64/nagios/plugins`）。
- *<NAGIOS_ETC_DIR>* - **Nagios**設定を含むディレクトリ（例：`/etc/nagios`）。
- *<NAGIOS_OBJECTS_DIR>* - **Nagios**オブジェクトを含むディレクトリ（例：`/etc/nagios/objects`）。

- *<NAGIOSGRAPH_DIR>* - **Nagiosgraph**ディレクトリ（例：`/usr/local/nagiosgraph`）。
- *<NAGIOS_PERFDATA_LOG>* - **Nagios**がサービスチェックの結果を記録するファイル（*<NAGIOSGRAPH_DIR>*`/etc/nagiosgraph.conf`の`perflog`ファイルと同じでなければなりません）。このファイルのレコードは*<NAGIOSGRAPH_DIR>*`/bin/insert.pl`スクリプトによって読み取られ、対応するRRAアーカイブ**RRDツール**に記録されます。

Nagiosを設定する：

1. `check_drweb`ファイルを*<NAGIOS_PLUGINS_DIR>*ディレクトリに、`drweb.cfg`ファイルを*<NAGIOS_OBJECTS_DIR>*ディレクトリにコピーします。
2. 監視対象のDr.Web for UNIX Internet Gatewaysがあるホストを*drweb*グループに追加します。ホストではDr.Web SNMPDが実行されている必要があります。デフォルトでは、このグループには*localhost*のみが追加されます。
3. 必要に応じて、**snmpIwalk**ツールを介して*drweb*ホストのDr.Web SNMPDに接続するように指示する`check_drweb`コマンドを編集します。

   ```
   snmpwalk -c public -v 2c $HOSTADDRESS$:161
   ```

   SNMPプロトコルの適切なバージョンとパラメータ（「*コミュニティストリング*」や認証パラメータなど）の他、ポートを指定します。`$HOSTADDRESS$`変数をコマンドに含める必要があります（この変数は後でコマンドが呼び出されたときに、**Nagios**によって正しいホストアドレスに自動的に置き換えられるためです）。このコマンドではOIDは必要ありません。また、実行ファイルへのフルパス（通常は`/usr/local/bin/snmpwalk`）を使用してコマンドを指定することをお勧めします。
4. 次の文字列をファイルに追加して、*<NAGIOS_ETC_DIR>*`/nagios.cfg`設定ファイルの*DrWeb*オブジェクトを接続します。

   ```
   cfg_file= <NAGIOS_OBJECTS_DIR>/drweb.cfg
   ```

5. *DrWeb*グラフィック用の**RRDツール**設定を`rrdopts.conf-sample`ファイルから*<NAGIOSGRAPH_DIR>*`/etc/rrdopts.conf`ファイルに追加します。
6. **Nagiosgraph**がまだ設定されていない場合は、その設定に対して次の手順を実行します。
   - `nagiosgraph.cfg`ファイルを*<NAGIOS_OBJECTS_DIR>*ディレクトリにコピーし、**process-service-perfdata-for-nagiosgraph**コマンドで`insert.pl`スクリプトへのパスを編集します。たとえば、次のようになります。

     ```
     $ awk '$1 == "command_line" { $2 = "<NAGIOSGRAPH_DIR>/bin/insert.pl" }
     { print }' ./objects/nagiosgraph.cfg >
     <NAGIOS_OBJECTS_DIR>/nagiosgraph.cfg
     ```

   - 次の行を追加して、*<NAGIOS_ETC_DIR>*`/nagios.cfg`設定ファイルにこのファイルを接続します。

     ```
     cfg_file=<NAGIOS_OBJECTS_DIR>/nagiosgraph.cfg
     ```

7. *<NAGIOS_ETC_DIR>*`/nagios.cfg`設定ファイルの**Nagios**パラメータの値を確認します。

```
check_external_commands=1
execute_host_checks=1
accept_passive_host_checks=1
enable_notifications=1
enable_event_handlers=1

process_performance_data=1
service_perfdata_file=/usr/nagiosgraph/var/rrd/perfdata.log
service_perfdata_file_template=$LASTSERVICECHECK$||$HOSTNAME$||$SERVICEDE
SC$||$SERVICEOUTPUT$||$SERVICEPERFDATA$
service_perfdata_file_mode=a
service_perfdata_file_processing_interval=30
service_perfdata_file_processing_command=process-service-perfdata-for-
nagiosgraph

check_service_freshness=1
enable_flap_detection=1
enable_embedded_perl=1
enable_environment_macros=1
```

NagiosのSNMPトラップ通知の受信を設定する

1. モニタリング対象ホストのDr.Web SNMPD設定（**TrapReceiver**パラメータ）で、**Nagios**が動作しているホストで**snmptrapd**が待ち受け（リッスン）するアドレスを指定します。次に例を示します。

   ```
   SNMPD.TrapReceiver = 10.20.30.40:162
   ```

2. *SNMPトラップ*を受信したときに呼び出される *<NAGIOS_PLUGINS_DIR>*/eventhandlers/submit_check_resultスクリプトがあるかどうかを確認します。スクリプトが見つからない場合は、*<opt_dir>*/share/drweb-snmpd/connectors/nagios/plugins/eventhandlers/ディレクトリのsubmit_check_resultファイルをこの場所にコピーします。このファイルで、CommandFileパラメータに指定されているパスを変更します。これは、*<NAGIOS_ETC_DIR>*/nagios.cfgファイルのcommand_fileパラメータと同じ値である必要があります。

3. snmptt.drweb.nagios.confファイルを/etc/snmp/snmp/ディレクトリにコピーします。このファイルで、パスをsubmit_check_resultに変更します。たとえば、次のようなコマンドを使用します。

   ```
   $ awk '$1 == "EXEC" { $2 =
   <NAGIOS_PLUGINS_DIR>/eventhandlers/submit_check_result }{ print}'
   ./snmp/snmptt.drweb.nagios.conf > /etc/snmp/snmp/snmptt.drweb.nagios.conf
   ```

4. 「/etc/snmp/snmptt.drweb.nagios.conf」文字列を/etc/snmp/snmptt.iniファイルに追加します。その後、デーモンモードで起動している場合は**snmptt**を再起動します。

Nagiosに必要な設定ファイルをすべて追加して編集したら、次のコマンドを使用して**Nagios**をデバッグモードで実行します。

```
# nagios -v <NAGIOS_ETC_DIR>/nagios.cfg
```

このコマンドを受け取ると、**Nagios**は設定エラーをチェックします。エラーがなかった場合は、**Nagios**を通常どおりに再起動できます（たとえば、**service** nagios restartコマンドを使用して）。

Nagiosの公式マニュアルについては、http://www.nagios.org/documentation/を参照してください。

Dr.Web SNMP MIB

SNMPプロトコルを介して外部モニタリングシステムから取得できるDr.Web for UNIX Internet Gatewaysの動作パラメータの一覧を以下の表に示します。

パラメータ名	パラメータのOID	パラメータのタイプと説明
すべての名前に共通のプレフィックス: .iso.org.dod.internet.private.enterprises.drweb.drwebSnmpd すべてのOIDに共通のプレフィックス: .1.3.6.1.4.1.29690.2		
alert	**イベントに関する非同期通知（SNMPトラップ）**	
threatAlert	.1.1	脅威の検出に関する通知
threatAlertFile	.1.1.1	感染ファイル名（文字列）
threatAlertType	.1.1.2	脅威の種類（整数*）
threatAlertName	.1.1.3	脅威名（文字列）
threatAlertOrigin	.1.1.4	脅威を検出したコンポーネントの識別子（整数***）
threatAlertRemoteIp	.1.1.5	ファイルへのアクセスに使用されたリモートホストのIPアドレス（文字列）
threatAlertRemoteUser	.1.1.6	ファイルにアクセスしたリモートユーザーの名前（文字列）
threatAlertRemoteDomain	.1.1.7	ファイルへのアクセスに使用されたリモートホストの名前（文字列）
threatActionErrorAlert	.1.2	脅威を駆除しようとしたときに発生したエラーに関する通知
threatActionErrorAlertFile	.1.2.1	感染ファイル名（文字列）
threatActionErrorAlertType	.1.2.2	脅威の種類（整数*）
threatActionErrorAlertName	.1.2.3	脅威名（文字列）
threatActionErrorAlertOrigin	.1.2.4	脅威を検出したコンポーネントの識別子（整数***）
threatActionErrorAlertError	.1.2.5	エラーの説明（文字列）
threatActionErrorAlertErrorCode	.1.2.6	エラーコード（エラーカタログのコードに対応する整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
threatActionErrorAlertAction	.1.2.7	失敗したアクション（1 - 修復、2 - 隔離、3 - 削除、4 - 報告、5 - 無視）
componentFailureAlert	.1.3	コンポーネント障害に関する通知
componentFailureAlertName	.1.3.1	コンポーネント識別子（整数***）
componentFailureAlertExitCodeDescription	.1.3.2	コンポーネント終了コードの説明（文字列）
componentFailureAlertExitCode	.1.3.3	エラーコード（エラーカタログのコードに対応する整数）
infectedUrlAlert	.1.4	悪質なURLのブロックに関する通知（HTTP/HTTPS接続）
infectedUrlAlertUrl	.1.4.1	ブロックされたURL（文字列）
infectedUrlAlertDirection	.1.4.2	HTTPメッセージの方向（整数：1 - 要求、2 - 応答）
infectedUrlAlertType	.1.4.3	脅威の種類（整数*）
infectedUrlAlertName	.1.4.4	脅威名（文字列）
infectedUrlAlertOrigin	.1.4.5	脅威を検出したコンポーネントの識別子（整数***）
infectedUrlAlertSrcIp	.1.4.6	接続元のIPアドレス（文字列）
infectedUrlAlertSrcPort	.1.4.7	接続元のポート（整数）
infectedUrlAlertDstIp	.1.4.8	接続先のIPアドレス（文字列）
infectedUrlAlertDstPort	.1.4.9	接続先のポート（整数）
infectedUrlAlertSniHost	.1.4.10	接続先のSNI（SSL接続）（文字列）
infectedUrlAlertExePath	.1.4.11	接続を確立したプログラムの実行可能パス（文字列）
infectedUrlAlertUserName	.1.4.12	接続を確立するプログラムを実行する権限を持つユーザーの名前（文字列）
infectedAttachmentAlert	.1.5	感染したメールの添付ファイルの検出に関する通知
infectedAttachmentAlertType	.1.5.1	脅威の種類（整数*）
infectedAttachmentAlertName	.1.5.2	脅威名（文字列）

パラメータ名	パラメータのOID	パラメータのタイプと説明
infectedAttachmentAlertOrigin	.1.5.3	脅威を検出したコンポーネントの識別子（整数***）
infectedEmailAttachmentAlertSocket	.1.5.4	メールメッセージの送信元のIPアドレス（文字列）
infectedEmailAttachmentAlertMailFrom	.1.5.5	メールメッセージの送信者（文字列）
infectedEmailAttachmentAlertRcptTo	.1.5.6	メールメッセージの受信者（文字列）
infectedEmailAttachmentAlertMessageId	.1.5.7	メールメッセージのMessage-IDヘッダーの値（文字列）
infectedEmailAttachmentAlertAction	.1.5.8	メールメッセージ全体または感染した添付ファイルに適用されたアクション（整数：1 - リパック、2 - 拒否、3 - 破棄、4 - 修復、5 - 隔離、6 - 削除）
infectedEmailAttachmentAlertDivert	.1.5.9	メールメッセージの方向（整数：1 - 受信、2 - 送信）
infectedEmailAttachmentAlertSrcIp	.1.5.10	接続元のIPアドレス（文字列）
infectedEmailAttachmentAlertSrcPort	.1.5.11	接続元のポート（整数）
infectedEmailAttachmentAlertDstIp	.1.5.12	接続先のIPアドレス（文字列）
infectedEmailAttachmentAlertDstPort	.1.5.13	接続先のポート（整数）
infectedEmailAttachmentAlertSniHost	.1.5.14	接続先のSNI（SSL接続）（文字列）
infectedEmailAttachmentAlertProtocol	.1.5.15	プロトコルの種類（整数：1 - SMTP、2 - POP3、3 - IMAP、4 - HTTP）
infectedEmailAttachmentAlertExePath	.1.5.16	接続を確立したプログラムの実行可能パス（文字列）
infectedEmailAttachmentAlertUserName	.1.5.17	接続を確立するプログラムを実行する権限を持つユーザーの名前（文字列）
categoryUrlAlert	.1.6	不要なカテゴリーに属するURLのブロックに関する通知
categoryUrlAlertUrl	.1.6.1	ブロックされたURL（文字列）
categoryUrlAlertCategory	.1.6.2	URLが属するWebリソースカテゴリー（整数**）

パラメータ名	パラメータのOID	パラメータのタイプと説明
categoryUrlAlertOrigin	.1.6.3	脅威を検出したコンポーネントの識別子（整数***）
categoryUrlAlertSrcIp	.1.6.4	接続元のIPアドレス（文字列）
categoryUrlAlertSrcPort	.1.6.5	接続元のポート（整数）
categoryUrlAlertDstIp	.1.6.6	接続先のIPアドレス（文字列）
categoryUrlAlertDstPort	.1.6.7	接続先のポート（整数）
categoryUrlAlertSniHost	.1.6.8	接続先のSNI（SSL接続）（文字列）
categoryUrlAlertExePath	.1.6.9	接続を確立したプログラムの実行可能パス（文字列）
categoryUrlAlertUserName	.1.6.10	接続を確立するプログラムを実行する権限を持つユーザーの名前（文字列）
categoryUrlEmailAttachmentAlert	.1.7	メールメッセージ内の不要なURLの検出に関する通知
categoryUrlEmailAttachmentAlertType	.1.7.1	URLが属するWebリソースカテゴリー（整数**）
categoryUrlEmailAttachmentAlertOrigin	.1.7.2	脅威を検出したコンポーネントの識別子（整数***）
categoryUrlEmailAttachmentAlertSocket	.1.7.3	メールメッセージの送信元のIPアドレス（文字列）
categoryUrlEmailAttachmentAlertMailFrom	.1.7.4	メールメッセージの送信者（文字列）
categoryUrlEmailAttachmentAlertRcptTo	.1.7.5	メールメッセージの受信者（文字列）
categoryUrlEmailAttachmentAlertMessageId	.1.7.6	メールメッセージのMessage-IDヘッダーの値（文字列）
categoryUrlEmailAttachmentAlertAction	.1.7.7	メールメッセージ全体または添付ファイルに適用されたアクション（整数：1 - リパック、2 - 拒否、3 - 破棄、4 - 修復、5 - 隔離、6 - 削除）
categoryUrlEmailAttachmentAlertDivert	.1.7.8	メールメッセージの方向（整数：1 - 受信、2 - 送信）
categoryUrlEmailAttachmentAlertSrcIp	.1.7.9	接続元のIPアドレス（文字列）

パラメータ名	パラメータのOID	パラメータのタイプと説明
categoryUrlEmailAttachmentAlertSrcPort	.1.7.10	接続元のポート(整数)
categoryUrlEmailAttachmentAlertDstIp	.1.7.11	接続先のIPアドレス(文字列)
categoryUrlEmailAttachmentAlertDstPort	.1.7.12	接続先のポート(整数)
categoryUrlEmailAttachmentAlertSniHost	.1.7.13	接続先のSNI(SSL接続)(文字列)
categoryUrlEmailAttachmentAlertProtocol	.1.7.14	プロトコルの種類(整数:1 - SMTP、2 - POP3、3 - IMAP、4 - HTTP)
categoryUrlEmailAttachmentAlertExePath	.1.7.15	接続を確立したプログラムの実行可能パス(文字列)
categoryUrlEmailAttachmentAlertUserName	.1.7.16	接続を確立するプログラムを実行する権限を持つユーザーの名前(文字列)
spamEmailAlert	.1.8	メールメッセージのスパム認識に関する通知
spamEmailAlertOrigin	.1.8.1	脅威を検出したコンポーネントの識別子(整数***)
spamEmailAlertSocket	.1.8.2	メールメッセージの送信元のIPアドレス(文字列)
spamEmailAlertMailFrom	.1.8.3	メールメッセージの送信者(文字列)
spamEmailAlertRcptTo	.1.8.4	メールメッセージの受信者(文字列)
spamEmailAlertMessageId	.1.8.5	メールメッセージのMessage-IDヘッダーの値(文字列)
spamEmailAlertAction	.1.8.6	メールメッセージ全体または添付ファイルに適用されたアクション(整数:1 - リパック、2 - 拒否、3 - 破棄、4 - 修復、5 - 隔離、6 - 削除)
spamEmailAlertDivert	.1.8.7	メールメッセージの方向(整数:1 - 受信、2 - 送信)
spamEmailAlertSrcIp	.1.8.8	接続元のIPアドレス(文字列)
spamEmailAlertSrcPort	.1.8.9	接続元のポート(整数)
spamEmailAlertDstIp	.1.8.10	接続先のIPアドレス(文字列)

パラメータ名	パラメータのOID	パラメータのタイプと説明
spamEmailAlertDstPort	.1.8.11	接続先のポート（整数）
spamEmailAlertSniHost	.1.8.12	接続先のSNI（SSL接続）（文字列）
spamEmailAlertProtocol	.1.8.13	プロトコルの種類（整数：1 - SMTP、2 - POP3、3 - IMAP、4 - HTTP）
spamEmailAlertExePath	.1.8.14	接続を確立したプログラムの実行可能パス（文字列）
spamEmailAlertUserName	.1.8.15	接続を確立するプログラムを実行する権限を持つユーザーの名前（文字列）
blockedConnectionAlert	.1.9	ネットワーク接続のブロックに関する通知
blockedConnectionAlertOrigin	.1.9.1	脅威を検出したコンポーネントの識別子（整数***）
blockedConnectionAlertDivert	.1.9.2	接続の方向（整数：1 - 受信、2 - 送信）
blockedConnectionAlertSrcIp	.1.9.3	接続元のIPアドレス（文字列）
blockedConnectionAlertSrcPort	.1.9.4	接続元のポート（整数）
blockedConnectionAlertDstIp	.1.9.5	接続先のIPアドレス（文字列）
blockedConnectionAlertDstPort	.1.9.6	接続先のポート（整数）
blockedConnectionAlertSniHost	.1.9.7	接続先のSNI（SSL接続）（文字列）
blockedConnectionAlertProtocol	.1.9.8	プロトコルの種類（整数：1 - SMTP、2 - POP3、3 - IMAP、4 - HTTP）
blockedConnectionAlertExePath	.1.9.9	接続を確立したプログラムの実行可能パス（文字列）
blockedConnectionAlertUserName	.1.9.10	接続を確立するプログラムを実行する権限を持つユーザーの名前（文字列）
stat	**Dr.Web for UNIX Internet Gatewaysの動作に関する統計**	
threatCounters	.2.1	検出された脅威のカウンター

パラメータ名	パラメータのOID	パラメータのタイプと説明
knownVirus	.2.1.1	検出された既知のウイルスの数（カウンター、整数）
suspicious	.2.1.2	検出された疑わしいオブジェクトの数（カウンター、整数）
adware	.2.1.3	検出されたアドウェアの数（カウンター、整数）
dialers	.2.1.4	検出されたダイアラーの数（カウンター、整数）
joke	.2.1.5	検出されたジョークプログラムの数（カウンター、整数）
riskware	.2.1.6	検出されたリスクウェアの数（カウンター、整数）
hacktool	.2.1.7	検出されたハッキングツールの数（カウンター、整数）
scanErrors	.2.2	ファイルのスキャン中に発生したエラーのカウンター
sePathNotAbsolute	.2.2.1	「パスが絶対パスではありません」エラーの発生回数（カウンター、整数）
seFileNotFound	.2.2.2	「ファイルが見つかりません」エラーの発生回数（カウンター、整数）
seFileNotRegular	.2.2.3	「ファイルは通常のファイルではありません」エラーの発生回数（カウンター、整数）
seFileNotBlockDevice	.2.2.4	「ファイルはブロックデバイスではありません」エラーの発生回数（カウンター、整数）
seNameTooLong	.2.2.5	「パスまたはファイル名が長すぎます」エラーの発生回数（カウンター、整数）
seNoAccess	.2.2.6	「パーミッションが拒否されました」エラーの発生回数（カウンター、整数）
seReadError	.2.2.7	「読み取りエラー」の発生回数（カウンター、整数）
seWriteError	.2.2.8	「書き込みエラー」の発生回数（カウンター、整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
seFileTooLarge	.2.2.9	「ファイルサイズが大きすぎます」エラーの発生回数（カウンター、整数）
seFileBusy	.2.2.10	「ファイルがビジーです」エラーの発生回数（カウンター、整数）
seUnpackingError	.2.2.20	「アンパックエラー」の発生回数（カウンター、整数）
sePasswordProtecetd	.2.2.21	「パスワード保護」エラーの発生回数（カウンター、整数）
seArchCrcError	.2.2.22	「アーカイブのCRCエラー」の発生回数（カウンター、整数）
seArchInvalidHeader	.2.2.23	「無効なアーカイブヘッダーです」エラーの発生回数（カウンター、整数）
seArchNoMemory	.2.2.24	「アーカイブを処理するのに十分なメモリがありません」エラーの発生回数（カウンター、整数）
seArchIncomplete	.2.2.25	「不完全なアーカイブ」エラーの発生回数（カウンター、整数）
seCanNotBeCured	.2.2.26	「オブジェクトを修復できません」エラーの発生回数（カウンター、整数）
sePackerLevelLimit	.2.2.30	「パックされたオブジェクトが最大ネストレベルを超えました」エラーの発生回数（カウンター、整数）
seArchiveLevelLimit	.2.2.31	「アーカイブが最大ネストレベルを超えました」エラーの発生回数（カウンター、整数）
seMailLevelLimit	.2.2.32	「メールファイルが最大ネストレベルを超えました」エラーの発生回数（カウンター、整数）
seContainerLevelLimit	.2.2.33	「コンテナが最大ネストレベルを超えました」エラーの発生回数（カウンター、整数）
seCompressionLimit	.2.2.34	「アーカイブの最大圧縮率を超えました」エラーの発生回数（カウンター、整数）
seReportSizeLimit	.2.2.35	「スキャン結果レポートの最大サイズを超えました」エラーの発生回数

パラメータ名	パラメータのOID	パラメータのタイプと説明
		（カウンター、整数）
seScanTimeout	.2.2.40	「スキャンタイムアウトに達しました」エラーの発生回数（カウンター、整数）
seEngineCrash	.2.2.41	「Scan Engineのクラッシュが検出されました」エラーの発生回数（カウンター、整数）
seEngineHangup	.2.2.42	「Scan Engineが応答を停止しました」エラーの発生回数（カウンター、整数）
seEngineError	.2.2.44	「Scan Engineの内部エラー」の発生回数（カウンター、整数）
seNoLicense	.2.2.45	「有効なライセンスが見つかりません」エラーの発生回数（カウンター、整数）
seCuringLimitReached	.2.2.47	「修復試行限界に達しました」エラーの発生数（カウンター、整数）
seNonSupportedDisk	.2.2.50	「サポートされていないディスクです」エラーの発生回数（カウンター、整数）
seUnexpectedError	.2.2.100	「予期せぬエラーです」の発生数（カウンター、整数）
scanLoadAverage	.2.3	ファイルスキャン負荷の指標
filesScannedTable	.2.3.1	他のコンポーネントの要求によりスキャンされたファイルの平均数
filesScannedEntry	.2.3.1.1	製品のコンポーネント（テーブル行全体、レコード）
filesScannedIndex	.2.3.1.1.1	コンポーネントのインデックス（識別子、整数***）
filesScannedOrigin	.2.3.1.1.2	コンポーネントの名前
filesScanned1min	.2.3.1.1.3	1秒間にチェックされたファイルの平均（1分間の平均）数（文字列）
filesScanned5min	.2.3.1.1.4	1秒間にチェックされたファイルの平均（5分間の平均）数（文字列）
filesScanned15min	.2.3.1.1.5	1秒間にチェックされたファイルの平均（15分間の平均）数（文字列）

パラメータ名	パラメータのOID	パラメータのタイプと説明
bytesScannedTable	.2.3.2	他のコンポーネントの要求により実行されたスキャンの平均速度（1秒あたりのバイト数）
bytesScannedEntry	.2.3.2.1	製品のコンポーネント（テーブル行全体、レコード）
bytesScannedIndex	.2.3.2.1.1	コンポーネントのインデックス（識別子、整数***）
bytesScannedOrigin	.2.3.2.1.2	コンポーネントの名前
bytesScanned1min	.2.3.2.1.3	1秒間にスキャンされた平均バイト数（1分間の平均）（文字列）
bytesScanned5min	.2.3.2.1.4	1秒間にスキャンされた平均バイト数（5分間の平均）（文字列）
bytesScanned15min	.2.3.2.1.5	1秒間にスキャンされた平均バイト数（15分間の平均）（文字列）
cacheHitFilesTable	.2.3.3	コンポーネントの要求によりキャッシュから取得されたスキャンレポートの平均数
cacheHitFilesEntry	.2.3.3.1	製品のコンポーネント（テーブル行全体、レコード）
cacheHitFilesIndex	.2.3.3.1.1	コンポーネントのインデックス（識別子、整数***）
cacheHitFilesOrigin	.2.3.3.1.2	コンポーネントの名前
cacheHitFiles1min	.2.3.3.1.3	1秒間にキャッシュから取得されたレポートの平均（1分間の平均）数（文字列）
cacheHitFiles5min	.2.3.3.1.4	1秒間にキャッシュから取得されたレポートの平均（5分間の平均）数（文字列）
cacheHitFiles15min	.2.3.3.1.5	1秒間にキャッシュから取得されたレポートの平均（15分間の平均）数（文字列）
errorsTable	.2.3.4	コンポーネントの要求により実行されたスキャン中の平均エラー数
errorsEntry	.2.3.4.1	製品のコンポーネント（テーブル行全体、レコード）
errorsIndex	.2.3.4.1.1	コンポーネントのインデックス（識別子、整数***）

パラメータ名	パラメータのOID	パラメータのタイプと説明
errorsOrigin	.2.3.4.1.2	コンポーネントの名前
errors1min	.2.3.4.1.3	1秒間のスキャンエラーの平均（1分間の平均）数（文字列）
errors5min	.2.3.4.1.4	1秒間のスキャンエラーの平均（5分間の平均）数（文字列）
errors15min	.2.3.4.1.5	1秒間のスキャンエラーの平均（15分間の平均）数（文字列）
net	.2.4	ネットワークアクティビティの統計
markedAsSpam	.2.4.1	スパムとしてマークされたメールメッセージの数（カウンター、整数）
blockedInfectionSource	.2.4.101	「感染源」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedNotRecommended	.2.4.102	「非推奨」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedAdultContent	.2.4.103	「アダルトコンテンツ」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedViolence	.2.4.104	「暴力」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedWeapons	.2.4.105	「武器」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedGambling	.2.4.106	「ギャンブル」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedDrugs	.2.4.107	「麻薬」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedObsceneLanguage	.2.4.108	「卑猥な表現」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedChats	.2.4.109	「チャット」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedTerrorism	.2.4.110	「テロリズム」カテゴリーに属するブロックされたURLの数（カウンター、整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
blockedFreeEmail	.2.4.111	「無料メールサービス」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedSocialNetworks	.2.4.112	「ソーシャルネットワーク」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedOwnersNotice	.2.4.113	「著作権者の申し立て」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedOnlineGames	.2.4.114	「オンラインゲーム」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedAnonymizers	.2.4.115	「アノニマイザー」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedCryptocurrencyMiningPools	.2.4.116	「仮想通貨マイニングプール」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedJobs	.2.4.117	「求人検索サイト」カテゴリーに属するブロックされたURLの数（カウンター、整数）
blockedBlackList	.2.4.120	ユーザーのブラックリストからブロックされたURLの数（カウンター、整数）
info	**Dr.Web for UNIX Internet Gatewaysの現在の状態に関する情報**	
components	.3.1	Dr.Web for UNIX Internet Gatewaysコンポーネントのステータス
configd	.3.1.1	**drweb-configd**コンポーネントデータ
configdState	.3.1.1.1	コンポーネントの現在の状態（整数****）
configdExitCode	.3.1.1.2	前回の終了コード（エラーカタログのコードに対応する整数）
configdExitTime	.3.1.1.3	前回の終了時刻（*UNIX時間*）
configdInstalledApps	.3.1.1.101	インストールされたコンポーネントのリスト

パラメータ名	パラメータのOID	パラメータのタイプと説明
configdAppEntry	.3.1.1.101.1	インストールされているコンポーネントに関する情報（テーブル行全体、レコード）
configdAppIndex	.3.1.1.101.1.1	インストールされているコンポーネントのインデックス（序数）（整数）
configdAppName	.3.1.1.101.1.2	インストールされているコンポーネントの名前（文字列）
configdAppExePath	.3.1.1.101.1.3	コンポーネントの実行ファイルへのパス（文字列）
configdAppInstallTime	.3.1.1.101.1.4	コンポーネントがインストールされた時刻（*UNIX時間*）
configdAppIniSection	.3.1.1.101.1.5	設定ファイルにあるコンポーネントのパラメータを含むセクションの名前
scanEngine	.3.1.2	**drweb-se**コンポーネントデータ
scanEngineState	.3.1.2.1	コンポーネントの現在の状態（整数****）
scanEngineExitCode	.3.1.2.2	前回の終了コード（エラーカタログのコードに対応する整数）
scanEngineExitTime	.3.1.2.3	前回の終了時刻（*UNIX時間*）
scanEngineStatus	.3.1.2.101	Dr.Web Virus-Finding Engineの現在の状態（整数）
scanEngineVersion	.3.1.2.102	Dr.Web Virus-Finding Engineのバージョン（文字列）
scanEngineVirusRecords	.3.1.2.103	ウイルスレコードの数（整数）
scanEngineMaxForks	.3.1.2.104	スキャン対象の子プロセスの最大数（整数）
scanEngineQueues	.3.1.2.105	スキャンタスクのキュー
scanEngineQueuesLow	.3.1.2.105.1	優先度の低いタスクのキュー
scanEngineQueueLowOut	.3.1.2.105.1.1	キューから出て処理に転送された優先度の低いタスクの数（カウンター、整数）
scanEngineQueueLowSize	.3.1.2.105.1.2	キュー内で処理待ちになっている優先度の低いタスクの数（カウンター、整数）
scanEngineQueuesMedium	.3.1.2.105.2	通常優先度のタスクのキュー

パラメータ名	パラメータのOID	パラメータのタイプと説明
scanEngineQueueMediumOut	.3.1.2.105.2.1	キューから出て処理に転送された通常優先度のタスクの数（カウンター、整数））
scanEngineQueueMediumSize	.3.1.2.105.2.2	キュー内で処理待ちになっている通常優先度のタスクの数（カウンター、整数）
scanEngineQueuesHigh	.3.1.2.105.3	優先度の高いタスクのキュー
scanEngineQueueHighOut	.3.1.2.105.3.1	キューから出て処理に転送された優先度の高いタスクの数（カウンター、整数））
scanEngineQueueHighSize	.3.1.2.105.3.2	キュー内で処理待ちになっている優先度の高いタスクの数（カウンター、整数）
scanEngineVirusBasesTable	.3.1.2.106	ウイルスデータベースのリスト。
scanEngineVirusBasesEntry	.3.1.2.106.1	ウイルスデータベースに関する情報（テーブル行全体、レコード）
scanEngineVirusBaseIndex	.3.1.2.106.1.1	ウイルスデータベースのインデックス（整数）
scanEngineVirusBasePath	.3.1.2.106.1.2	ウイルスデータベースディレクトリへのパス（文字列）
scanEngineVirusBaseRecords	.3.1.2.106.1.3	ウイルスデータベース内のレコード数（整数）
scanEngineVirusBaseVersion	.3.1.2.106.1.4	ウイルスデータベースのバージョン（整数）
scanEngineVirusBaseTimestamp	.3.1.2.106.1.5	ウイルスデータベースのタイムスタンプ（*UNIX時間*）
scanEngineVirusBaseMD5	.3.1.2.106.1.6	MD5チェックサム（文字列）
scanEngineVirusBaseLoadResult	.3.1.2.106.1.7	このウイルスデータベースのダウンロード結果（文字列）
scanEngineQueuesTab	.3.1.2.107	スキャンタスクキューのリスト
scanEngineQueueEntry	.3.1.2.107.1	キューに関する情報（テーブル行全体、レコード）
scanEngineQueueIndex	.3.1.2.107.1.1	キューのインデックス（序数）（整数）
scanEngineQueueName	.3.1.2.107.1.2	キューの名前（文字列）

パラメータ名	パラメータのOID	パラメータのタイプと説明
scanEngineQueueOut	.3.1.2.107.1.3	キューから出て処理に転送されたタスクの数（カウンター、整数）
scanEngineQueueSize	.3.1.2.107.1.4	キュー内で処理待ちになっているタスクの数（カウンター、整数）
fileCheck	.3.1.3	**drweb-filecheck**コンポーネントデータ
fileCheckState	.3.1.3.1	コンポーネントの現在の状態（整数****）
fileCheckExitCode	.3.1.3.2	前回の終了コード（エラーカタログのコードに対応する整数）
fileCheckExitTime	.3.1.3.3	前回の終了時刻（*UNIX時間*）
fileCheckScannedFiles	.3.1.3.101	スキャン済みファイル数（カウンター、整数）
fileCheckScannedBytes	.3.1.3.102	スキャン済みバイト数（カウンター、整数）
fileCheckCacheHitFiles	.3.1.3.103	キャッシュから取得したスキャンレポートの数（カウンター、整数）
fileCheckScanErrors	.3.1.3.104	Scan Engineでのエラーの発生回数（カウンター、整数）
fileCheckScanStat	.3.1.3.105	クライアントのリスト
fileCheckClientEntry	.3.1.3.105.1	クライアントに関する情報（テーブル行全体、レコード）
fileCheckClientIndex	.3.1.3.105.1.1	クライアントのインデックス（序数）（整数）
fileCheckClientName	.3.1.3.105.1.2	クライアントコンポーネントの名前（文字列）
fileCheckClientScannedFiles	.3.1.3.105.1.3	このクライアントでスキャンされたファイル数（カウンター、整数）
fileCheckClientScannedBytes	.3.1.3.105.1.4	このクライアントでスキャンされたバイト数（カウンター、整数）
fileCheckClientCacheHitFiles	.3.1.3.105.1.5	このクライアントのキャッシュから取得したスキャンレポートの数（カウンター、整数）
fileCheckClientScanErrors	.3.1.3.105.1.6	このクライアントで動作しているときにScan Engineで発生したエラーの数（カウンター、整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
update	.3.1.4	**drweb-update**コンポーネントデータ
updateState	.3.1.4.1	コンポーネントの現在の状態（整数****）
updateExitCode	.3.1.4.2	前回の終了コード（エラーカタログのコードに対応する整数）
updateExitTime	.3.1.4.3	前回の終了時刻（*UNIX時間*）
updateBytesSent	.3.1.4.101	送信したバイト数（カウンター、整数）
updateBytesReceived	.3.1.4.102	受信したバイト数（カウンター、整数）
esagent	.3.1.5	**drweb-esagent**コンポーネントデータ
esagentState	.3.1.5.1	コンポーネントの現在の状態（整数****）
esagentExitCode	.3.1.5.2	前回の終了コード（エラーカタログのコードに対応する整数）
esagentExitTime	.3.1.5.3	前回の終了時刻（*UNIX時間*）
esagentWorkStatus	.3.1.5.101	コンポーネントの現在の動作モード（整数：1 - スタンドアロンモード、2 - 接続中、3 - 接続待ち、4 - 接続承認済み）
esagentIsConnected	.3.1.5.102	サーバーに接続されているかどうか（整数、0 - いいえ、1 - はい）
esagentServer	.3.1.5.103	使用されている集中管理サーバーのアドレス（文字列）
netcheck	.3.1.6	**drweb-netcheck**コンポーネントデータ
netcheckState	.3.1.6.1	コンポーネントの現在の状態（整数****）
netcheckExitCode	.3.1.6.2	前回の終了コード（エラーカタログのコードに対応する整数）
netcheckExitTime	.3.1.6.3	前回の終了時刻（*UNIX時間*）
netcheckLocalSeForks	.3.1.6.101	ローカルで利用可能なScan Engineプロセスの数（整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
netcheckRemoteSeForks	.3.1.6.102	リモートで利用可能なScan Engineプロセスの数（整数）
netcheckLocalFilesScanned	.3.1.6.103	ローカルでスキャンされたファイルの数（カウンター、整数）
netcheckNetworkFilesScanned	.3.1.6.104	リモートスキャンでスキャンされたファイルの数（カウンター、整数）
netcheckLocalBytesScanned	.3.1.6.105	ローカルでスキャンされたバイト数（カウンター、整数）
netcheckNetworkBytesScanned	.3.1.6.106	リモートスキャンでスキャンされたバイト数（カウンター、整数）
netcheckLocalBytesIn	.3.1.6.107	ローカルクライアントから受信したバイト数（カウンター、整数）
netcheckLocalBytesOut	.3.1.6.108	ローカルクライアントに送信したバイト数（カウンター、整数）
netcheckNetworkBytesIn	.3.1.6.109	リモートホストから受信したバイト数（カウンター、整数）
netcheckNetworkBytesOut	.3.1.6.110	リモートホストに送信したバイト数（カウンター、整数）
netcheckLocalScanErrors	.3.1.6.111	ローカルのScan Engineプロセスでのエラーの発生回数（カウンター、整数）
netcheckNetworkScanErrors	.3.1.6.112	リモートのScan Engineプロセスでのエラーの発生回数（カウンター、整数）
httpd	.3.1.7	**drweb-httpd**コンポーネントデータ
httpdState	.3.1.7.1	コンポーネントの現在の状態（整数****）
httpdExitCode	.3.1.7.2	前回の終了コード（エラーカタログのコードに対応する整数）
httpdExitTime	.3.1.7.3	前回の終了時刻（*UNIX時間*）
snmpd	.3.1.8	**drweb-snmpd**コンポーネントデータ
snmpdState	.3.1.8.1	コンポーネントの現在の状態（整数****）
snmpdExitCode	.3.1.8.2	前回の終了コード（エラーカタログのコードに対応する整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
snmpdExitTime	.3.1.8.3	前回の終了時刻（*UNIX時間*）
clamd	.3.1.20	**drweb-clamd**コンポーネントデータ
clamdState	.3.1.20.1	コンポーネントの現在の状態（整数****）
clamdExitCode	.3.1.20.2	前回の終了コード（エラーカタログのコードに対応する整数）
clamdExitTime	.3.1.20.3	前回の終了時刻（*UNIX時間*）
icapd	.3.1.21	**drweb-icapd**コンポーネントデータ
icapdState	.3.1.21.1	コンポーネントの現在の状態（整数****）
icapdExitCode	.3.1.21.2	前回の終了コード（エラーカタログのコードに対応する整数）
icapdExitTime	.3.1.21.3	前回の終了時刻（*UNIX時間*）
icapdConnectionsIn	.3.1.21.101	承認された受信接続の数（カウンター、整数）
icapdConnectionsCount	.3.1.21.102	現在開いている接続の数（カウンター、整数）
icapdOptions	.3.1.21.103	*OPTIONS*要求の数（カウンター、整数）
icapdReqmod	.3.1.21.104	*REQMOD*要求の数（カウンター、整数）
icapdRespmod	.3.1.21.105	*RESPMOD*要求の数（カウンター、整数）
icapdBad	.3.1.21.106	無効なリクエストの数（カウンター、整数）
smbspider	.3.1.40	**drweb-smbspider-daemon**コンポーネントデータ
smbspiderState	.3.1.40.1	コンポーネントの現在の状態（整数****）
smbspiderExitCode	.3.1.40.2	前回の終了コード（エラーカタログのコードに対応する整数）
smbspiderExitTime	.3.1.40.3	前回の終了時刻（*UNIX時間*）
smbspiderConnectionsIn	.3.1.40.101	開かれた接続の合計数（カウンター、整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
smbspiderConnectionsCount	.3.1.40.102	現在開いている接続の数（カウンター、整数）
smbspiderShareTable	.3.1.40.103	保護された**Samba**共有リソースに関する統計
smbspiderShareEntry	.3.1.40.103.1	保護されている**Samba**共有リソースに関する情報（テーブル行全体、レコード）
smbspiderShareIndex	.3.1.40.103.1.1	保護された**Samba**共有リソースのインデックス（序数）（整数）
smbspiderSharePath	.3.1.40.103.1.2	保護された**Samba**共有リソースへのパス（文字列）
smbspiderShareConnectionsIn	.3.1.40.103.1.3	開かれた接続の合計数（カウンター、整数）
smbspiderShareConnectionsCount	.3.1.40.103.1.4	現在開いている接続の数（カウンター、整数）
gated	.3.1.41	**drweb-gated**コンポーネントデータ
gatedState	.3.1.41.1	コンポーネントの現在の状態（整数****）
gatedExitCode	.3.1.41.2	前回の終了コード（エラーカタログのコードに対応する整数）
gatedExitTime	.3.1.41.3	前回の終了時刻（*UNIX時間*）
gatedInterceptedIn	.3.1.41.101	モニタリングした接続の数（カウンター、整数）
gatedInterceptedCount	.3.1.41.102	現在モニタリングされている接続の数（カウンター、整数）
maild	.3.1.42	**drweb-maild**コンポーネントデータ
maildState	.3.1.42.1	コンポーネントの現在の状態（整数****）
maildExitCode	.3.1.42.2	前回の終了コード（エラーカタログのコードに対応する整数）
maildExitTime	.3.1.42.3	前回の終了時刻（*UNIX時間*）
maildStat	.3.1.42.4	**drweb-maild**コンポーネントによる処理の統計
maildStatNative	.3.1.42.4.1	コンポーネントの内部インターフェース**drweb-maild**を介したメールス

パラメータ名	パラメータのOID	パラメータのタイプと説明
		キャンの統計（傍受したSMTP、POP3、IMAP接続のスキャン中にSpIDer Gateが受信したメッセージ）
maildStatNativePassed	.3.1.42.4.1.1	受信できなかったメッセージの数（カウンター、整数）
maildStatNativeRepacked	.3.1.42.4.1.2	再パッケージ化されたメッセージの数（カウンター、整数）
maildStatNativeRejected	.3.1.42.4.1.3	拒否されたメッセージの数（カウンター、整数）
maildStatNativeFailed	.3.1.42.4.1.4	メッセージスキャンエラーの数（カウンター、整数）
maildStatNativeQueueSize	.3.1.42.4.1.5	キューライン、つまりインターフェースを介してのスキャンを待機しているファイルの数（整数）
maildStatMilter	.3.1.42.4.2	**drweb-maild**コンポーネントのコンポーネントインターフェース*Milter*を介したメールスキャンの統計
maildStatMilterPassed	.3.1.42.4.2.1	受信できなかったメッセージの数（カウンター、整数）
maildStatMilterRepacked	.3.1.42.4.2.2	再パッケージ化されたメッセージの数（カウンター、整数）
maildStatMilterRejected	.3.1.42.4.2.3	拒否されたメッセージの数（カウンター、整数）
maildStatMilterFailed	.3.1.42.4.2.4	メッセージスキャンエラーの数（カウンター、整数）
maildStatMilterQueueSize	.3.1.42.4.2.5	キューライン、つまりインターフェースを介してのスキャンを待機しているファイルの数（整数）
maildStatSpamc	.3.1.42.4.3	**drweb-maild**コンポーネントのコンポーネントインターフェース*Spamd*を介したメールスキャンの統計
maildStatSpamcPassed	.3.1.42.4.3.1	受信できなかったメッセージの数（カウンター、整数）
maildStatSpamcRepacked	.3.1.42.4.3.2	再パッケージ化されたメッセージの数（カウンター、整数）
maildStatSpamcRejected	.3.1.42.4.3.3	拒否されたメッセージの数（カウンター、整数）

パラメータ名	パラメータのOID	パラメータのタイプと説明
maildStatSpamcFailed	.3.1.42.4.3.4	メッセージスキャンエラーの数（カウンター、整数）
maildStatSpamcQueueSize	.3.1.42.4.3.5	キューライン、つまりインターフェースを介してのスキャンを待機しているファイルの数（整数）
maildStatRspamc	.3.1.42.4.4	**drweb-maild**コンポーネントのコンポーネントインターフェース*Rspamd*を介したメールスキャンの統計
maildStatRspamcPassed	.3.1.42.4.4.1	受信できなかったメッセージの数（カウンター、整数）
maildStatRspamcRepacked	.3.1.42.4.4.2	再パッケージ化されたメッセージの数（カウンター、整数）
maildStatRspamcRejected	.3.1.42.4.4.3	拒否されたメッセージの数（カウンター、整数）
maildStatRspamcFailed	.3.1.42.4.4.4	メッセージスキャンエラーの数（カウンター、整数）
maildStatRspamcQueueSize	.3.1.42.4.4.5	キューライン、つまりインターフェースを介してのスキャンを待機しているファイルの数（整数）
lookupd	.3.1.43	**drweb-lookupd**コンポーネントデータ
lookupdState	.3.1.43.1	コンポーネントの現在の状態（整数****）
lookupdExitCode	.3.1.43.2	前回の終了コード（エラーカタログのコードに対応する整数）
lookupdExitTime	.3.1.43.3	前回の終了時刻（*UNIX時間*）
antispam	.3.1.44	**drweb-ase**コンポーネントに関するデータ
antispamState	.3.1.44.1	コンポーネントの現在の状態（整数****）
antispamExitCode	.3.1.44.2	前回の終了コード（エラーカタログのコードに対応する整数）
antispamExitTime	.3.1.44.3	前回の終了時刻（*UNIX時間*）
cloudd	.3.1.50	**drweb-cloudd**コンポーネントデータ

パラメータ名	パラメータのOID	パラメータのタイプと説明
clouddState	.3.1.50.1	コンポーネントの現在の状態（整数****）
clouddExitCode	.3.1.50.2	前回の終了コード（エラーカタログのコードに対応する整数）
clouddExitTime	.3.1.50.3	前回の終了時刻（*UNIX時間*）
vpnd	.3.1.51	**drweb-vpnd**コンポーネントデータ
vpndState	.3.1.51.1	コンポーネントの現在の状態（整数****）
vpndExitCode	.3.1.51.2	前回の終了コード（エラーカタログのコードに対応する整数）
vpndExitTime	.3.1.51.3	前回の終了時刻（*UNIX時間*）
vpndWorkStatus	.3.1.51.101	コンポーネントの現在の動作モード（整数：0 - オフ、1 - サーバー、2 - クライアント）
vpndConnectionState	.3.1.51.102	確立された接続のステータス（整数：0 - ステータス未設定、1 - 接続中、2 - 接続済み、3 - エラー、4 - NATの設定中、5 - 保護トンネルの作成中）
vpndNetworkName	.3.1.51.103	作成したパーソナルネットワークの名前（文字列）
meshd	.3.1.52	**drweb-meshd**コンポーネントデータ
meshdState	.3.1.52.1	コンポーネントの現在の状態（整数****）
meshdExitCode	.3.1.52.2	前回の終了コード（エラーカタログのコードに対応する整数）
meshdExitTime	.3.1.52.3	前回の終了時刻（*UNIX時間*）
lotus	.3.1.60	**drweb-lotus**コンポーネントデータ
lotusState	.3.1.60.1	コンポーネントの現在の状態（整数****）
lotusExitCode	.3.1.60.2	前回の終了コード（エラーカタログのコードに対応する整数）
lotusExitTime	.3.1.60.3	前回の終了時刻（*UNIX時間*）

パラメータ名	パラメータのOID	パラメータのタイプと説明
macgui	.3.1.100	**drweb-gui**（**macOS**用）コンポーネントデータ
macguiState	.3.1.100.1	コンポーネントの現在の状態（整数****）
macguiExitCode	.3.1.100.2	前回の終了コード（エラーカタログのコードに対応する整数）
macguiExitTime	.3.1.100.3	前回の終了時刻（*UNIX時間*）
macspider	.3.1.102	**drweb-spider**（**macOS**用）コンポーネントデータ
macspiderState	.3.1.102.1	コンポーネントの現在の状態（整数****）
macspiderExitCode	.3.1.102.2	前回の終了コード（エラーカタログのコードに対応する整数）
macspiderExitTime	.3.1.102.3	前回の終了時刻（*UNIX時間*）
macspiderWorkStatus	.3.1.102.101	コンポーネントの現在の動作モード（整数：0 - 未設定、1 - 読み込み中、2 - 実行中）
macfirewall	.3.1.103	**drweb-firewall**（**macOS**用）コンポーネントデータ
macfirewallState	.3.1.103.1	コンポーネントの現在の状態（整数****）
macfirewallExitCode	.3.1.103.2	前回の終了コード（エラーカタログのコードに対応する整数）
macfirewallExitTime	.3.1.103.3	前回の終了時刻（*UNIX時間*）
linuxgui	.3.1.200	**drweb-gui**（**GNU/Linux**の場合）コンポーネントデータ
linuxguiState	.3.1.200.1	コンポーネントの現在の状態（整数****）
linuxguiExitCode	.3.1.200.2	前回の終了コード（エラーカタログのコードに対応する整数）
linuxguiExitTime	.3.1.200.3	前回の終了時刻（*UNIX時間*）
linuxspider	.3.1.201	**drweb-spider**（**GNU/Linux**の場合）コンポーネントデータ
linuxspiderState	.3.1.201.1	コンポーネントの現在の状態（整数****）

パラメータ名	パラメータのOID	パラメータのタイプと説明
linuxspiderExitCode	.3.1.201.2	前回の終了コード（エラーカタログのコードに対応する整数）
linuxspiderExitTime	.3.1.201.3	前回の終了時刻（*UNIX時間*）
linuxspiderWorkStatus	.3.1.201.101	コンポーネントの現在の動作モード（整数：0 - 未設定、1 - 読み込み中、2 - **fanotify**経由で実行中、3 - LKM経由で実行中）
linuxnss	.3.1.202	**drweb-nss**（**GNU/Linux**の場合）コンポーネントデータ
linuxnssState	.3.1.202.1	コンポーネントの現在の状態（整数****）
linuxnssExitCode	.3.1.202.2	前回の終了コード（エラーカタログのコードに対応する整数）
linuxnssExitTime	.3.1.202.3	前回の終了時刻（*UNIX時間*）
linuxnssScannedFiles	.3.1.202.101	スキャン済みファイル数（カウンター、整数）
linuxnssScannedBytes	.3.1.202.102	スキャン済みバイト数（カウンター、整数）
linuxnssScanErrors	.3.1.202.103	スキャンエラーの発生回数（カウンター、整数）
linuxfirewall	.3.1.203	**drweb-firewall**（**GNU/Linux**の場合）コンポーネントデータ
linuxfirewallState	.3.1.203.1	コンポーネントの現在の状態（整数****）
linuxfirewallExitCode	.3.1.203.2	前回の終了コード（エラーカタログのコードに対応する整数）
linuxfirewallExitTime	.3.1.203.3	前回の終了時刻（*UNIX時間*）
ctl	.3.1.300	**drweb-ctl**コンポーネントデータ
ctlState	.3.1.300.1	コンポーネントの現在の状態（整数****）
ctlExitCode	.3.1.300.2	前回の終了コード（エラーカタログのコードに対応する整数）
ctlExitTime	.3.1.300.3	前回の終了時刻（*UNIX時間*）
license	.3.2	ライセンスのステータス

パラメータ名	パラメータのOID	パラメータのタイプと説明
licenseEsMode	.3.2.1	ライセンスは集中管理サーバーによって許可されました（整数：0 - いいえ、1 - はい）
licenseNumber	.3.2.2	ライセンス番号（整数）
licenseOwner	.3.2.3	ライセンスの所有者（文字列）
licenseActivated	.3.2.4	ライセンスを有効化した日（*UNIX時間*）
licenseExpires	.3.2.5	ライセンス有効期限（*UNIX時間*）

*) 脅威のタイプ:

コード	脅威の種類
1	既知のウイルス（*known virus*）
2	疑わしいオブジェクト（*suspicious*）
3	アドウェア（*adware*）
4	ダイアラー（*dialer*）
5	ジョークプログラム（*joke program*）
6	リスクウェア（*riskware*）
7	ハッキングツール（*hacktool*）

**) URLのカテゴリー:

コード	脅威の種類
1	感染源（*infectionSource*）
2	非推奨（*notRecommended*）
3	アダルトコンテンツ（*adultContent*）
4	暴力（*violence*）
5	武器（*weapons*）
6	ギャンブル（*gambling*）
7	麻薬（*drugs*）

コード	脅威の種類
8	卑猥な表現など(*obsceneLanguage*)
9	チャット(*chats*)
10	テロリズム(*terrorism*)
11	無料メール(*freeEmail*)
12	ソーシャルネットワーク(*socialNetworks*)
13	著作権者からの申し立てによって登録されたURL(*ownerNotice*)
14	オンラインゲーム(*onlineGames*)
15	アノニマイザー(*anonymizers*)
16	暗号通貨マイニングプール(*cryptocurrencyMiningPools*)
17	求人検索サイト(*Jobs*)
20	ブラックリストに追加済み(*blackList*)

***) Dr.Webコンポーネントのコード:

コード	コンポーネント
1	Dr.Web ConfigD (**drweb-configd**)
2	Dr.Web Scanning Engine (**drweb-se**)
3	Dr.Web File Checker (**drweb-filecheck**)
4	Dr.Web Updater (**drweb-update**)
5	Dr.Web ES Agent (**drweb-esagent**)
6	Dr.Web Network Checker (**drweb-netcheck**)
7	Dr.Web HTTPD (**drweb-httpd**)
8	Dr.Web SNMPD (**drweb-snmpd**)
20	Dr.Web ClamD (**drweb-clamd**)
21	Dr.Web ICAPD (**drweb-icapd**)
40	SpIDer Guard for SMB (**drweb-smbspider-daemon**)
41	SpIDer Gate (**drweb-gated**)

コード	コンポーネント
42	Dr.Web MailD (**drweb-maild**)
43	Dr.Web LookupD (**drweb-lookupd**)
50	Dr.Web CloudD (**drweb-cloudd**)
51	Dr.Web VPND (**drweb-vpnd**)
52	Dr.Web MeshD (**drweb-meshd**)
60	Dr.Web for **Lotus**
100	**drweb-gui** for **macOS**
102	SpIDer Guard for macOS for **macOS**
103	Dr.Web Firewall for macOS for **macOS**
200	**drweb-gui** for **GNU/Linux**
201	SpIDer Guard (**drweb-spider**)
202	SpIDer Guard for NSS (**drweb-nss**)
203	Dr.Web Firewall for Linux（**drweb-firewall**）for **GNU/Linux**
300	Dr.Web Ctl (**drweb-ctl**)
400	Enterprise scanner（Dr.Web for UNIX Internet Gatewaysのコンポーネントではありません）

****) コンポーネントの状態：

コード	ステータス
0	インストールされていません
1	インストールされているが開始されていない
2	起動中
3	実行中
4	終了中

変数の値を直接取得するには、**snmpwalk**ユーティリティを使用します。

```
$ snmpwalk -Os -c <community> -v <SNMP version> <host address> <OID>
```

たとえば、ローカルホストで検出された脅威に関する統計を取得するには、次のコマンドを使用します（Dr.Web SNMPDの設定がデフォルト値に設定されている場合）。

```
$ snmpwalk -Os -c public -v 2c 127.0.0.1 .1.3.6.1.4.1.29690.2.2.1
```

Dr.Web MeshD

Dr.Web MeshDは、「ローカルクラウド」にインストールされたDr.Web for UNIX Internet Gatewaysを持つホストを含むエージェントです。この製品は、ホストをインストール済みのDr.Web for UNIX製品に接続します。このクラウドを利用すると、次のタスクを解決できます。

- 複数のクラウドホストによる他のファイルスキャンサービスの提供（スキャンコアサービス）。
- ウイルスデータベースの更新をクラウドホスト間で配布する。

ホストをインストール済みのDr.Web for UNIX製品に接続するには、Dr.Web MeshDコンポーネントをすべてのホストにインストールする必要があります。コンポーネントによってクラウドのホストが組み込まれます。クラウド内のホストの権限とホストが使用するクラウド機能は、Dr.Web MeshD設定で簡単に設定できます。

データは、保護されたSSHチャネルを介して他のクラウドホストと共有されます。

動作原理

Dr.Web MeshDは、Dr.Web for UNIX Internet Gatewaysがインストールされているホストと他のクラウドホスト間のインタラクションを調整します。

- 接続タイプ
- 動作モード
- サービス

接続タイプ

Dr.Web MeshDでは、次の接続タイプを使用します。

- *クライアント（サービス）* - Dr.Web MeshDが他のクラウドホストに接続するために使用されます。これらのホストは、指定したホストによって提供されるサービスのクライアントです。

  ホスト上で動作し、同じホスト上で動作するDr.Web MeshDを介してクラウド提供のサービスにアクセスするDr.Web for UNIX Internet Gatewaysのコンポーネントは、ローカルのUNIXソケットを介してクライアントに接続します。その場合、クライアント接続は使用されません。

- *パートナー（ピアツーピア）* - ピア（サービス内）パートナークラウドホストとのインタラクションのためにDr.Web MeshDによって使用されます。通常、このような接続は、クラウドとやり取りするときの負荷の拡張と分散の他、クラウドホストの同期に使用されます。
- *アップリンク* - このホスト（クライアント）をクラウドホスト（サービスプロバイダー）に接続する際にDr.Web MeshDによって使用されますスキャンのためのファイル転送など）。

使用する3つの種類の接続はすべて、それぞれのクラウドサービスに対して個別に設定されます。さらに同じホストを、サービス内でクライアントの要求（最新の更新を提供するなど）を処理するサーバーとして、および別のサービス内（リモートファイルスキャンなど）のクライアントとして設定できます。

クラウド内では、ホストは安全なSSHプロトコルを介して承認済みのインタラクションを実行します。つまり、ホスト間通信のすべての側面が常に相互認証されます。認証には、RFC 4251に従って*ホストキー*が使用されます。ローカルコンポーネントからのクライアント接続は常に信頼済みと見なされます。

動作モード

Dr.Web MeshDは、デーモンモードで動作し、ローカルホストにある他のDr.Web for UNIX Internet Gatewaysコンポーネントからの要求で実行できます。Dr.Web MeshDがクライアント接続を提供するように設定されており（つまり、**ListenAddress**パラメータが空でない場合）、少なくとも1つのサービスがアクティブ化されている場合、Dr.Web MeshDはデーモンとして起動し、クライアントからの接続を待機します。また、Dr.Web MeshDは、リクエストに応じて、たとえば次のコマンドを実行するときに、ローカルホストで有効化できます。

```
$ drweb-ctl update --local-cloud
```

Dr.Web MeshDがクライアント接続を処理するように設定されておらず（**ListenAddress**パラメータが空の場合）、**IdleTimeLimit**パラメータで指定された期間中にDr.Web MeshDへのリクエストがない場合、コンポーネントは自動的に終了します。

サービス

更新を交換する（更新）

このサービスを使用すると、ホストはウイルスやその他のデータベースの更新をサブスクライブし、更新に関する通知を送信し、クラウドホスト間で更新ファイルをアップロードして共有できます。サービス設定は、**Update***パラメータを使用して設定できます。

標準のサービス使用法では、Dr.Web MeshDがインストールされ、会社のローカルネットワーク内の複数のマシン（サービスのクライアント）で更新を取得する機能が有効になっていると想定しています。一般的なクライアント設定は次のとおりです。

```
…
[MeshD]
UpdateChannel = On
UpdateUplink = <server address>
ListenAddress =
…
[Update]
UseLocalCloud = Yes
…
```

更新を配布するローカルサーバーでは、次の設定が指定されています。

```
UpdateChannel = On
UpdateUplink =
ListenAddress = <address>:<port>
```

ここで、クライアントのアップリンク接続の*<server address>*は、サーバーホストがクライアント接続を管理するために使用する*<address>*および*<port>*を参照する必要があります。

ホストの1つが更新サーバー（ローカルクラウドの外部：Dr.Web GUS更新サーバーまたは集中管理サーバー）から更新されている場合、ホストは必要なすべてのクライアントに通知を送信し（ホストが更新交換サーバーとして設定されている場合）、サーバーホストに、ホストから配布可能なファイルの新しいリストを送信します。通知を受信すると、クライアントホストはサーバーからの更新のダウンロードをリクエストできます。次に、クライアントから

ファイルをリクエストしてローカルに保存するか、サーバーからファイルをリクエストした他のクライアントに送信できます。

このシナリオでは、クライアントが異なるタイミングでDr.Web GUSにリクエストを送信するため、更新の遅延が短縮します。そのため、最初に更新されたクライアントは、必要なすべてのクラウドホストに最新の更新ファイルをすぐに配信します。また、トラフィック量とDr.Web GUSの負荷も軽減されます。

ローカルクラウドを更新の配布に使用する場合、Dr.Web MeshDに加えて、ホストにはDr.Web Updaterコンポーネントが含まれている必要があることに注意してください。

リモートファイルスキャン（エンジン）

このサービスでは、Dr.Web Scanning Engineを使用してリモートファイルをスキャンできます。クライアントホストはスキャン用のファイルをサーバーホストに送信し、サーバーホストはファイルスキャン用のサービスを提供します。一般的なクライアント設定は次のとおりです。

```
…
[MeshD]
EngineChannel = On
EngineUplink = <server address>
ListenAddress =
…
```

ローカルスキャンサーバーでは、次の設定が指定されています。

```
EngineChannel = On
EngineUplink =
ListenAddress = <address>: <port>
```

ここで、クライアントのアップリンク接続の*<server address>*は、サーバーホストがクライアント接続を管理するために使用する*<address>*および*<port>*を参照する必要があります。

スキャンするファイルを送信する（ファイル）

この機能は使用されません（リモートスキャンは*Engine*サービス内で提供されます）。

URLチェック

このサービスでは、サーバーホストを使用して、潜在的に危険で推奨されないカテゴリーに属するURLをチェックできます。クライアントホストは、チェックするURLをサーバーホストに送信します。 一般的なクライアント設定は次のとおりです。

```
…
[MeshD]
UrlChannel = On
UrlUplink = <server address>
ListenAddress =
…
```

URLチェックに使用されるローカルサーバーでは、次の設定が指定されています。

```
UrlChannel = On
UrlUplink =
ListenAddress = <address>:<port>
```

ここで、クライアントのアップリンク接続の*<server address>*は、サーバーホストがクライアント接続を管理するために使用する*<address>*および*<port>*を参照する必要があります。

コマンドライン引数

Dr.Web MeshDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-meshd [<parameters>]
```

Dr.Web MeshDは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-meshd --help
```

このコマンドはDr.Web MeshDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。Dr.Web ConfigD設定デーモンによって必要になる場合は、自動的に起動されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

> コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-meshd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの[MeshD]セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-meshd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-meshd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-meshd
DebugSsh *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合、ホスト上で動作しているDr.Web MeshDが送受信したSSHプロトコルメッセージ（メッセージとデータの転送に使用）のロギングを実行。 **デフォルト値**：No
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 最小値 - 10s. **デフォルト値**：30s
DnsResolverConfPath *{path to file}*	DNSリゾルバ設定ファイルへのパス。 **デフォルト値**：/etc/resolv.conf
ListenAddress *<IP address>:<port>*	コンポーネントがクラウドホストからの接続の受信を待機しているクライアント接続のネットワークソケット（アドレスとポート）。これらのホストは、このクラウドホストによって提供されるサービスのクライアントです。 *値が指定されていない場合、コンポーネントはクライアントからの要求を受け取りません。* **デフォルト値**：*（未設定）*
UpdateChannel *{On / Off}*	このホスト上で動作するDr.Web MeshDコンポーネントを有効または無効にし、クラウドのホスト間でウイルスデータベースの更新を交換します（たとえば、他のクラウドホストからウイルスデータベースの更新を取得し、新しい更新をクラウドに送信します）。

	このパラメータがオンに設定されている場合、コンポーネントDr.Web MeshDはDr.Web ConfigD設定デーモンによって自動的に起動されます。 **デフォルト値**：On
UpdateUplink *{address}*	このホストに更新を提供するサーバーとして機能するDr.Web MeshDの上位ホストのアドレス。 **使用可能な値**： • *値が指定されていない* - サーバーがサービスに設定されておらず、Dr.Web MeshDはどこにも接続されていません。 • *<IP address>:<port>* - Dr.Web MeshDは指定されたアドレスとポートでサーバーに接続します。 • dns:*<service name>*[:*<domain>*] - サーバーのアドレスとポートは、*<domain>* DNSドメインのSRVレコードを検索することで指定されます。*<domain>*が指定されていない場合、DNSリゾルバ設定ファイルのドメインが使用されます（パスは**ResolverConfPath**で指定されます）。ドメインは、最後に検出されたドメインに応じて、searchまたはdomainフィールドから取得されます。 **デフォルト値**：*（指定なし）*
UpdateDebugIpc *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合は、更新の交換サービスのログにデバッグ情報を出力します。 **デフォルト値**：No
UpdateTraceContent *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合は、更新の交換サービスのログに送信済みデータを出力します。 **デフォルト値**：No
FileChannel *{On | Off}*	このホスト上で動作するDr.Web MeshDコンポーネントのファイル交換への参加を許可するオプションを有効または無効にします。 *このパラメータがオンに設定されている場合、コンポーネントDr.Web MeshDはDr.Web ConfigD設定デーモンによって自動的に起動されます。* **デフォルト値**：On
FileUplink *{address}*	このホストのファイルをスキャンするサーバーとして機能するDr.Web MeshDの上位ホストのアドレス。 **使用可能な値**： • *値が指定されていない* - サーバーがサービスに設定されておらず、Dr.Web MeshDはどこにも接続されていません。 • *<IP address>:<port>* - Dr.Web MeshDは指定されたアドレスとポートでサーバーに接続します。 • dns:*<service name>*[:*<domain>*] - サーバーのアドレスとポートは、*<domain>* DNSドメインのSRVレコードを検索することで指定されます。*<domain>*が指定されていない場合、DNSリゾルバ設定ファイルのドメインが使用されます（パスは**ResolverConfPath**で指定されます）。ドメインは、最後に検出されたドメインに応じて、searchまたはdomainフィールドから取得されます。 **デフォルト値**：*（指定なし）*

FileDebugIpc *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合は、ファイル交換サービスのログにデバッグ情報を出力します。 **デフォルト値**：No
EngineChannel *{On | Off}*	このホストで動作するDr.Web MeshDコンポーネントがスキャンエンジンサービスの提供に参加できるようにするオプションを有効または無効にします。 *このパラメータがオンに設定されている場合、コンポーネントDr.Web MeshDはDr.Web ConfigD設定デーモンによって自動的に起動されます。* **デフォルト値**：On
EngineUplink *{address}*	このホストにスキャンエンジンサービスを提供するスキャンサーバーとして機能するDr.Web MeshDの上位ホストのアドレス。 **使用可能な値**： • *値が指定されていない* - サーバーがサービスに設定されておらず、Dr.Web MeshDはどこにも接続されていません。 • *<IP address>:<port>* - Dr.Web MeshDは指定されたアドレスとポートでサーバーに接続します。 • dns:*<service name>*[:*<domain>*] - サーバーのアドレスとポートは、*<domain>* DNSドメインのSRVレコードを検索することで指定されます。*<domain>*が指定されていない場合、DNSリゾルバ設定ファイルのドメインが使用されます（パスは**ResolverConfPath**で指定されます）。ドメインは、最後に検出されたドメインに応じて、searchまたはdomainフィールドから取得されます。 **デフォルト値**：*（指定なし）*
EngineDebugIpc *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合は、スキャンサービスのログにデバッグ情報を出力します。 **デフォルト値**：No
UrlChannel *{On | Off}*	このホストで動作するDr.Web MeshDコンポーネントがURLチェックサービスの提供に参加できるようにするオプションを有効または無効にします。
UrlUplink *{address}*	このホストにURLチェックサービスを提供するサーバーとして機能するDr.Web MeshDの上位ホストのアドレス。 **使用可能な値**： • *値が指定されていない* - サーバーがサービスに設定されておらず、Dr.Web MeshDはどこにも接続されていません。 • *<IP address>:<port>* - Dr.Web MeshDは指定されたアドレスとポートでサーバーに接続します。 • dns:*<service name>*[:*<domain>*] - サーバーのアドレスとポートは、*<domain>* DNSドメインのSRVレコードを検索することで指定されます。*<domain>*が指定されていない場合、DNSリゾルバ設定ファイルのドメインが使用されます（パスは**ResolverConfPath**で指定されます）。ドメインは、最後に検出されたドメインに応じて、searchまたはdomainフィールドから取得されます。 **デフォルト値**：*（指定なし）*

UrlDebugIpc *{Boolean}*	ロギングレベルが**LogLevel** = Debugの場合、デバッグ情報をURLチェックサービスのログに出力します。 **デフォルト値**：No

Dr.Web for UNIX Internet Gatewaysの現在のバージョンでは、*File*ファイル送信サービスは使用されません。代わりに、*Engine*スキャンエンジンサービスを使用します。

Dr.Web CloudD

Dr.Web CloudDコンポーネントはDr.Web Cloud（Doctor Webのクラウドサービス）を参照します。Dr.Web Cloudサービスは、検出された脅威に関する最新情報をすべてのDr.Webアンチウイルスソリューションから収集し、ユーザーが望ましくないWebサイトにアクセスするのを防ぎ、Dr.Webウイルスデータベースに記述のない新しい脅威を含んだ感染ファイルからOSを保護します。さらに、Dr.Web Cloudサービスを使用すると、Dr.Web Scanning Engineスキャンエンジンとインターネットへのアクセスを監視するコンポーネントの誤検知の可能性が少なくなります。

動作原理

このコンポーネントは、指定ファイルのコンテンツにローカルのDr.Web Scanning Engineが把握していない脅威がないかスキャンし、指定URLがDoctor WebのWebリソースの定義済みカテゴリーに属するかどうかを確認するようDoctor Web Dr.Web Cloudサービスに指示します。また、コンポーネントは感染ファイルの検出に関する統計情報と、Dr.Web for UNIX Internet Gatewaysを実行しているOSに関する情報を定期的にDr.Web Cloudに送信します。

Dr.Web CloudDは設定デーモンによって自動的に実行されます。コンポーネントは、ユーザーまたはDr.Web for UNIX Internet Gatewaysコンポーネントの1つからコマンドを受信したときに実行されます。

このコンポーネントは、Dr.Web Cloudサービスにユーザーがリクエストしたurlのスキャンをリクエストする際に、ネットワークトラフィックのスキャンコンポーネント、SpIDer GateのURL、Dr.Web ICAPD。

さらに、コンポーネントはコマンドラインDr.Web Ctl（**drweb-ctl**コマンドによって起動）からのDr.Web for UNIX Internet Gateways管理ユーティリティのコマンドによるファイルのスキャン中に使用されます。脅威が検出されたら、Dr.Web Scanning Engineスキャンエンジンがファイルに関するレポートをDr.Web Cloudに送信します。

コマンドライン引数

Dr.Web CloudDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-cloudd [<parameters>]
```

Dr.Web CloudDは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-cloudd --help
```

このコマンドはDr.Web CloudDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理し、からの情報を使用してファイルやURLをスキャンするには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-cloudd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[CloudD]`セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル

	パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-cloudd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-cloudd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-cloudd
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。 **RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 **デフォルト値**：30s
PersistentCache *{On / Off}*	Dr.Web Cloud応答のキャッシュをディスクへ保存することを有効または無効にします。 **デフォルト値**：Off
DebugSdk *{Boolean}*	詳細なDr.Web Cloudメッセージをデバッグレベルでログファイルに含めるかどうかを示します（**LogLevel** = DEBUG）。 **デフォルト値**：No

Dr.Web LookupD

Dr.Web LookupDコンポーネントは、LDAPプロトコルを使用してデータを取得するために、外部ソース（テキストファイル、リレーショナルデータベース、ディレクトリサービス、LDAPプロトコルのインタラクションのサポート）を参照するように設計されています。受信したデータはルールで使用され、それに従ってネットワーク接続がスキャンされます（ユーザー認証のチェックなど）。このデータは、特定の条件が満たされた場合にURLへのアクセスをブロックするためにも使用されます。

コンポーネントの設定では、いくつかのデータソースに接続するためのパラメータを指定できます。Dr.Web LookupDは、Dr.Web for UNIX Internet Gatewaysのいずれかのコンポーネントからデータ要求を受信した場合にのみ、必要なデータソースに接続します。

Dr.Web LookupDでは、次のデータソースへの参照がサポートされます。

- テキストファイル（*AllMatch*、*Mask*、*Regex*、*Cidr*モードで）。
- リレーショナルデータベース（**MySQL**、**PostgreSQL**、**SQLite**）。
- **Redis**データストレージ。
- ディレクトリサービス（**Active Directory**や、LDAP経由のアクセスを提供するその他のサービス、たとえば**OpenLDAP**など）。

LDAPプロトコルによるデータの共有は、オープンチャネルまたは保護されたチャネル上でSSL/TLSを適用することで実行できます。安全な接続を使用するには、Dr.Web LookupDに適切なSSL証明書とキーを提供する必要があります。SSLキーと証明書を生成する必要がある場合は、**openssl**ユーティリティを使用できます。**openssl**ユーティリティを使用して証明書とプライベートキーを生成する方法の例については、付録E. SSL証明書を生成するセクションを参照してください。

動作原理

このコンポーネントは、テキストファイル、リレーショナルデータベース、ネットワークストレージ、LDAPプロトコルをサポートするディレクトリサービス（**Active Directory**など）にデータをリクエストするように設計されています。受信したデータ（ユーザーのIDや権限など）は、Dr.Web for UNIX Internet Gatewaysのコンポーネントに転送され、さまざまなルールでスキャンに使用されます（ユーザーがリクエストしたURLなどにアクセスを許可するなど）。

このマニュアルではリレーショナルデータベース、**Redis**ストレージ、ディレクトリサービス、LDAPプロトコルの動作原理については説明しません。必要に応じて、対応する参考資料を参照してください。

Dr.Web LookupDコンポーネントは、必要に応じて（データのリクエストを受信するときに）Dr.Web ConfigD設定デーモンによって自動的に起動されます。

特定のコンポーネントからのデータ受信に関するリクエストが到着すると、Dr.Web ConfigD構成デーモンがDr.Web LookupDを起動します（起動されていない場合）。次に、コンポーネントはリクエストされたデータソースからリクエストを実行してレスポンスを返します。リクエストに応じて、応答は所定の検索条件に従ってデータソースから取得された、特定の検索条件を満たす文字列のリスト、または所定の条件に一致する文字列が検索結果に含まれているかどうかを示す論理値（trueまたはfalse）で構成されます。

Dr.Web LookupD設定では、指定可能なデータソースの数に制限はありません。クライアントコンポーネントは、データ取得リクエストを作成するときに、データソースを指定する必要があります。Dr.Web LookupDは、起

動後しばらくの間、新しいリクエストを待機します。それ以上リクエストがない場合は、待機期間が経過した後、Dr.Web LookupDは自動的にシャットダウンします。

Dr.Web for UNIX Internet Gatewaysのその他のコンポーネントがDr.Web LookupDを使用する基本的な方法は、これらのコンポーネントの動作ルールで指定されているいくつかの条件の有効性を確認するために必要なデータを取得することです。ルールの適用性と条件の妥当性をチェックするときに、Dr.Web LookupDに対するデータリクエストが自動的に実行されます。

テキストファイル処理の特性

1. テキストファイルを処理するとき、文字列の先頭と末尾のスペースは破棄されます。空白行と最初の文字が「#」で始まる行は無視されます。
2. テキストファイルは不変のデータソースと見なされ、その内容はメモリ内に完全にキャッシュされます。さらに、検証のためのこれらのファイルに対するリクエストの結果もキャッシュされます。そのため、ソースファイルが変更されている場合は、**drweb-ctl**の`reload`コマンドなどを使用してDr.Web ConfigDコンポーネントにHUPシグナルを送信することで、Dr.Web LookupDに設定を再読み込みさせる必要があります。

MySQL接続の状況

MySQL接続の前に、**MySQL**ファイル設定の`[client]`セクションのパラメータがデフォルトで読み取られます（ファイル検索は、`/etc/my.cnf`、`/etc/mysql/my.cnf`、`/etc/alternatives/my.cnf`のパスで行われます）。

コマンドライン引数

Dr.Web LookupDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-lookupd [<parameters>]
```

Dr.Web LookupDは次のパラメータを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-lookupd --help
```

このコマンドはDr.Web LookupDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** `1 drweb-lookupd`

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの`[LookupD]`セクションで指定されている設定パラメータを使用します。

- コンポーネントパラメータ
- データソースセクション
- 新しいデータソース用のセクションを追加する

コンポーネントパラメータ

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、`[Root]`セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：`Notice`
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：`Auto`
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*`/bin/drweb-lookupd` • **GNU/Linux**の場合：`/opt/drweb.com/bin/drweb-lookupd` • **FreeBSD**の場合：`/usr/local/libexec/drweb.com/bin/drweb-lookupd`
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「`name:`」というプレフィックスを付けて指定します。次に例を示します。**RunAsUser** `= name:123456`。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。*

	デフォルト値：drweb
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 最小値 - 10s. **デフォルト値**：30s
DebugLibldap *{Boolean}*	**libldap**ライブラリのデバッグメッセージもデバッグレベルでログファイルに含めるかどうかを示します（つまり、**LogLevel** = DEBUGの場合）。 **デフォルト値**：No
LdapCheckCertificate *{No / Allow / Try / Yes}*	SSL/TLS経由のLDAP接続の証明書検証モード。 **使用可能な値**： • No - サーバーの証明書を要求しない。 • Allow - サーバーの証明書を要求する。証明書が提供されない場合、セッションは通常どおりに続行されます。サーバーの証明書が提供されたもののスキャンできない（対応するルート証明書が見つからない）場合、証明書は無視され、セッションは通常どおりに続行されます。 • Try - サーバーの証明書を要求する。証明書が提供されない場合、セッションは通常どおりに続行されます。サーバーの証明書が提供されたものの確認できない（対応するルート証明書が見つからない）場合、セッションは終了します。 • Yes - サーバーの証明書を要求する。証明書が提供されないか、スキャンできない場合（対応するルート証明書が見つからない場合）、セッションは終了します。 *LDAPデータソースの場合、この証明書検証モードは、*ldaps://*スキームまたはStartTLS拡張機能が使用されるときのURLの処理方法に影響します。*ADデータソースの場合は、*対応するセクション*で**UseSSL**=Yesが指定されている場合（以下参照）のサーバーへの接続に影響します。* **デフォルト値**：Yes
LdapCertificatePath *{path to file}*	安全なSSL/TLS接続を介したLDAPサーバーへの接続（**Active Directory**）に使用されるSSL証明書へのパス。 *証明書ファイルとプライベートキーファイル（後述のパラメータで指定されます）は、一致するペアを形成する必要があります。* **デフォルト値**：*（未設定）*
LdapKeyPath *{path to file}*	安全なSSL/TLS接続を介したLDAPサーバーへの接続（**Active Directory**）に使用されるプライベートキーへのパス。 *証明書ファイルとプライベートキーファイル（上のパラメータで指定されます）は、一致するペアを形成する必要があります。* **デフォルト値**：*（未設定）*
LdapCaPath *{path}*	SSL/TLSを介したLDAPプロトコルによるデータの共有に信頼して使用できる、信頼できるルート証明書のシステムリストを含むディレクトリまたはファイルへのパス。

	デフォルト値：*信頼できる証明書のリストへのパス。パスはお使いの* **GNU/Linux** *ディストリビューションに依存します。* • **Astra Linux**、**Debian**、**Linux Mint**、**SUSE Linux**、**Ubuntu** *の場合、通常は* `/etc/ssl/certs/` *です。* • **CentOS** *と* **Fedora** *の場合、* `/etc/pki/tls/certs/ca-bundle.crt` *です。* • *他のディストリビューションでは、コマンド* **openssl** `version -d` *の実行結果によってパスを定義できます。* • *コマンドが使用できない場合、またはOSディストリビューションを特定できない場合は、値* `/etc/ssl/certs/` *が使用されます。*
DbIdleTimeout *{time interval}*	データベース（または **Redis** ストレージ）への確立された接続がアイドル状態になっている場合にその接続を切断するまでのタイムアウト期間。 **デフォルト値**：`5m`
MysqlDefaultConn *{URL}*	**MySQL** データベースに接続するためのパラメータをデフォルトで設定するURL。 **使用可能な値**： • `tcp://[<user>[:<password>]@][<host>][:<port>][/<database name>][?<parameter>=<value>[&...]]` • `unix://[<user>[:<password>]@]<path to socket>[:<database name>][?<parameter>=<value>[&...]]` URI要件に注意してください。 **デフォルト値**：*（未設定）*
PqDefaultConn *{URL}*	**PostgreSQL** データベースに接続するためのパラメータをデフォルトで設定するURL。 **使用可能な値**： • `tcp://[<user>[:<password>]@][<host>][:<port>][/<database name>][?<parameter>=<value>[&...]]` • `unix://[<user>[:<password>]@]<path to socket>[:<database name>][?<parameter>=<value>[&...]]` URI要件に注意してください。 **デフォルト値**：*（未設定）*
SqliteDefaultConn *{path to file}*	デフォルトの **SQLite** データベースファイルへのパス（`file://` スキームプレフィックスを指定）。 **デフォルト値**：*（未設定）*
RedisDefaultConn *{URL}*	**PostgreSQL** データベースの接続パラメータをデフォルトで設定するURL。 **使用可能な値**： • `tcp://[<password>@][<host>][:<port>][/<database index>]` • `unix://[<password>@]<socket path>[:<database index>]` URI要件に注意してください。

	デフォルト値：*（未設定）*

データベース接続のURI要件

1. `tcp:`および`unix:`スキームプレフィックスのみを使用します（ローカルUNIXソケットの場合）。データベース固有のプレフィックス（`postgresql:`および`mysql:`など）はサポートしていません。**SQLite**データベースファイルへのパスは、`file://`スキームプレフィックスで指定されます。
2. *<host>*フィールドがURIで指定されていないか、`localhost`ホストが指定されている場合、ホストアドレス`127.0.0.1`が置き換えられます。この場合、**MySQL**および**PostgreSQL**データベースでは、ネットワーク接続が指定されているにもかかわらず、*ローカルUNIXソケットを介して*デフォルトで接続が確立されます。
3. URIフィールド（*<user>*、*<password>*、*<database name>*など）または接続パラメータ文字列に特殊文字（スペース、列など）が含まれている場合は、次のように16進コーディングを使用します。たとえば、
   - スペース -「%20」
   - ':' -「%3A」
   - '/' -「%2F」
   - '@' -「%40」
   - '%' -「%25」
4. **MySQL**の場合、接続パラメータ文字列には次のパラメータのみを含めることができます。

パラメータ名	データベースドキュメントの文字・記号	タイプ	説明
`init`	*MYSQL_INIT_COMMAND*	文字列	データベースへの接続後に実行するSQLコマンド
`compression`	*MYSQL_OPT_COMPRESS*	論理	データ圧縮を使用する
`connect-timeout`	*MYSQL_OPT_CONNECT_TIMEOUT*	整数	未使用の接続を切断するためのタイムアウト（秒）
`reconnect`	*MYSQL_OPT_RECONNECT*	論理	自動再接続を許可または拒否する
`read-timeout`	*MYSQL_OPT_READ_TIMEOUT*	整数	サーバーからのパケット受信のタイムアウト（秒）
`write-timeout`	*MYSQL_OPT_WRITE_TIMEOUT*	整数	パケットをサーバーに送信するためのタイムアウト（秒）
`charset`	*MYSQL_SET_CHARSET_NAME*	文字列	デフォルトの接続に使用される文字エンコーディングの名前
`plugin-dir`	*MYSQL_PLUGIN_DIR*	文字列	プラグインを格納するサーバーのディレクトリへのパス

パラメータ名	データベースドキュメントの文字・記号	タイプ	説明
nonblock	*MYSQL_OPT_NONBLOCK*	整数	ブロック以外のI/O操作のスタックサイズ
ssl-key	*MYSQL_OPT_SSL_KEY*	文字列	安全な接続を確立するために使用されるプライベートキー（PEM形式）へのパス
ssl-cert	*MYSQL_OPT_SSL_CERT*	文字列	安全な接続を確立するために使用されるパブリックキー証明書（PEM形式）へのパス
ssl-ca	*MYSQL_OPT_SSL_CA*	文字列	信頼できるCA証明書を含むファイル（PEM形式）へのパス
ssl-capath	*MYSQL_OPT_SSL_CAPATH*	文字列	信頼できるCA証明書を含むディレクトリへのパス（PEM形式）
ssl-cipher	*MYSQL_OPT_SSL_CIPHER*	文字列	安全な接続でサポートされている暗号化アルゴリズムのリスト
ssl-crl	*MYSQL_OPT_SSL_CRL*	文字列	失効した証明書を含むファイル（PEM形式）へのパス
ssl-crlpath	*MYSQL_OPT_SSL_CRLPATH*	文字列	失効した証明書を含むディレクトリへのパス（PEM形式）
ssl-fp	*MARIADB_OPT_SSL_FP*	文字列	有効なサーバー証明書のSHA1ハッシュ
ssl-fp-list	*MARIADB_OPT_SSL_FP_LIST*	文字列	有効なサーバー証明書のSHA1ハッシュを含むファイルへのパス
tls-passphrase	*MARIADB_OPT_TLS_PASSPHRASE*	文字列	パスワードで保護されたクライアントプライベートキーのパスワード
tls-version	*MARIADB_OPT_TLS_VERSION*	文字列	サポートされるTLSバージョンのリスト
server-verify-cert	*MYSQL_OPT_SSL_VERIFY_SERVER_CERT*	論理	サーバー証明書の検証を許可または禁止する
server-public-key-path	*MYSQL_SERVER_PUBLIC_KEY*	文字列	RSAサーバーパブリックキーを含むファイル

パラメータ名	データベースドキュメントの文字・記号	タイプ	説明
			（PEM形式）へのパス

データベースのドキュメントのパラメータの詳細については、https://mariadb.com/kb/en/library/mysql_optionsv/を参照してください。

5. **PostgreSQL**データベースについては、https://www.postgresql.org/docs/current/libpq-connect.html#LIBPQ-PARAMKEYWORDSも参照してください。

データソースセクション

設定ファイルには、一般的なセクションである`[LookupD]`の他、データソースへの接続を記述するセクション（各接続につき1つのセクション）もあります。これらのセクションには、`[LookupD.`*<タイプ>*`.`*<名前>*`]`というスキームを使用して名前が付けられます。ここで、

- *<タイプ>* - 接続の種類：
  - `LDAP` - **LDAP**を使用するディレクトリサービス用。
  - `AD` - **Active Directory**サービス。
  - `AllMatch` - *AllMatch*モードのテキストファイル用（フル認証）。
  - `Mask` - *マスク*モードのテキストファイル用（マスク認証）。
  - `Regex` - *Regex*モードのテキストファイル用（PCRE標準の正規表現に対する認証）。
  - `Cidr` - *Cidr*モードのテキストファイル用（IPアドレスまたはIPアドレス範囲認証）。
  - `Pq` - **PostgreSQL**データベース用。
  - `Mysql` - **Mysql**データベース用。
  - `Sqlite` - **Sqlite**データベース用。
  - `Redis` - **Redis**データベース用。
- *<名前>* - 接続の一意のID（タグ）。これを使用してルールから接続を参照できます。

例：`[LookupD.LDAP.auth1]`。データソースのセクション内に含まれるパラメータのセットは、接続の種類によって異なります。データソースセクションの数に制限はありません。

1.LDAPタイプのセクションで使用されるパラメータ

`Url` *{string}*	使用されるLDAPサーバーと抽出されるデータを定義するURL。URLは、RFC 4516に従って次のスキームに基づいて構築されます。 *<scheme>*`://`*<host>*`[:`*<port>*`]/`*<dn>*`[?`*<attrs>*`[?`*<scope>*`[?`*<filter>*`[?`*<extensions>*`]]]]` *<scheme>* - サーバーへの接続方法（許可されるスキーム：`ldap`、`ldaps`、`ldapi`）。 *<host>*`[:`*<port>*`]` - リクエストを受信するLDAPサーバーアドレス。 *<dn>* - オブジェクトの識別名。このオブジェクトに関する情報が送信されました。

	<attrs> - レコード属性の名前。この値はリクエスト内で受信する必要があります。 *<scope>* - 検索範囲（base、one、sub）。 *<filter>* - 抽出された属性の値に対するフィルター条件。 *<extensions>* - リクエストで使用されるLDAP拡張子のリスト。 **機能：** • 属性のリスト *<attrs>*では、「*」、「+」、「1.1」の任意の特殊文字を使用できます。 • 以下の自動的に解決されるプレースホルダーは、URLの *<dn>*および *<filter>*部分で使用できます。 ▫ $uは、クライアントコンポーネントによって送信されたユーザー名に自動的に置き換えられます。 ▫ $dは、クライアントコンポーネントによって送信されたドメインに自動的に置き換えられます。 ▫ $D - dc=*<subdomain>*、dc=*<domain>*に変更される *<subdomain>.<domain>*チェーン。 ▫ $$ -「$」文字。 • *<filter>*条件で特殊文字（「*」、「(」、「)」、「\」、コードが0の文字など）を通常の文字として使用する必要がある場合、それらの文字は\XXと記述する必要があります。さらに、URL LDAPの特殊文字は、シーケンス%XXを使用してエンコードされます。たとえば、「/」文字のldapiスキームに従ってURLをローカルLDAPサーバーソケットへのパスの一部として使用する場合、この文字は%2fとしてエンコードされます。 • *<extensions>*で許可される拡張機能として、StartTLSと1.3.6.1.4.1.1466.20037のみがサポートされます。これらの拡張機能では、保護されたスキームであるldapsを使用することが明示されていない場合でも、TLSメカニズム（つまり、LDAPサーバーとの保護された接続の確立）が使用されます。使用される拡張機能の名前の前に「!」文字がある場合は、TLSを使用する*必要があります*。つまり、安全な接続を確立できない場合には、リクエストは処理*されません*。それ以外の場合は、安全な接続が確立されていなくてもリクエストは処理されます。 指定された拡張機能は、保護されたldapsスキームでは使用できません。詳細については、RFC 4516または**man** ldap_search_ext_sを参照してください。 **例：** "ldaps://ds.example.com:990/$D?givenName,sn,cn?sub?(uid=$u)" "ldap://ldap.local/o=org,dc=nodomain?ipNetworkNumber?sub?(objectClass=ipNetwork)?!StartTLS" **デフォルト値：***（未設定）*
BindDn *{string}*	ユーザーが認証を受けなければならないLDAPディレクトリ内のオブジェクト。 **例：**「cn=admin,dc=nodomain」 **デフォルト値：***（未設定）*

BindPassword *{string}*	LDAPサーバー上での認証に使用するユーザーのパスワード。 **デフォルト値**：*（未設定）*
ChaseReferrals *{Boolean}*	現在のLDAPサーバーがリクエストへの応答として他のLDAPサーバーへの参照を返す場合、その参照に従うようコンポーネントに指示します。 **デフォルト値**：No

2.ADタイプのセクションで使用されるパラメータ

Host *{string}*	接続する**Active Directory**サービスのサーバーが稼働しているホストのドメイン名（FQDN）またはIPアドレス。 **例**：「win2012.win.local」 **デフォルト値**：*（未設定）*
Port *{integer}*	**Active Directory**サービスのサーバーが待ち受け（リッスン）するホスト上のポート。 **デフォルト値**：389
Dn *{string}*	**Active Directory**内のオブジェクトの*DN*。これはLDAP URLのdn部分に似ています。 **例**：「dc = win、dc = local」 **デフォルト値**：*（未設定）*
User *{string}*	識別に使用される、サーバー上のユーザーID。 **例**：「Administrator@WIN.LOCAL」 **デフォルト値**：*（未設定）*
Password *{string}*	**Active Directory**サーバー上での認証に使用されるユーザーのパスワード。 **デフォルト値**：*（未設定）*
ChaseReferrals *{Boolean}*	現在の**Active Directory**サーバーがリクエストへの応答として他のLDAPサーバーへの参照を返す場合、その参照に従うようコンポーネントに指示します。 **デフォルト値**：No
UseSSL *{Boolean}*	**Active Directory**サーバーへの接続にSSL/TLSを使用するよう指示します。 **デフォルト値**：No

3.AllMatch、Mask、Regex、Cidrタイプのセクションパラメータ

File *{path}*	検索文字列を含むテキストファイルへのパス。 **例**：「/etc/file1」 **デフォルト値**：*（未設定）*

注意

- AllMatchタイプのセクションで指定したファイルの文字列は、大文字と小文字を区別しない完全に一致する文字列の検索に使用されます。
- Maskタイプのセクションで指定したファイルの文字列は、マスク(*ワイルドカード*)と見なされます。マスクは、標準文字と特殊文字を含む正規表現の簡略版と見なすことができます。文字列とマスクの照合は、大文字と小文字を区別しないで行われます。マスクには、次の特殊文字と式を含めることができます。

  * - 任意の文字シーケンス。

  ? - 任意の1つの記号。

  [*<character set>*] - セットの文字(たとえば、[bac])

  [*<character set>*] - セットのどの記号にも一致しない文字(たとえば、[!cab])

  [[:*<class>*:]] - 文字(*alnum*、*alpha*、*ascii*、*blank*、*cntrl*、*digit*、*graph*、*lower*、*print*、*punct*、*space*、*upper*、*xdigit*)のPOSIXクラスの文字。

  サブストリングと一致するマスクには、「*」記号で囲まれたサブストリングが含まれている必要があります。(例:*host*)。いずれかの特殊文字を指定する必要がある場合は、バックスラッシュを使用して文字をエスケープする必要があります(\[、\]、*、\?)。必要に応じて、バックスラッシュをエスケープすることもできます(\\)。他の文字をエスケープしても意味はありません。たとえば文字列\a\b\c*\d\?\\は文字列abc*d?\に変換されます。マスクの例:

```
#「name」文字列に完全に一致します
name

#3文字の文字列に一致します:
#最初の文字は「c」、2番目は任意、3番目は「t」
#例:「cat」、「cut」、「cct」
c?t

#「user」、「users」、「us3rr」、「ussr1」などの文字列に一致します
#([:alpha:]文字クラスは任意のアルファベット
#文字、特殊文字「?」は任意の文字に一致します)
us[[:alpha:]34]r?

#「.con」、「file.col」、「3 ... co!」などの文字列に一致します
#(「.co」の前の任意の文字シーケンス、その後ろは
#「m」と「?」以外の任意の文字)
*.co[!m\?]

#「host」を含む任意の文字列に一致します。
#例:「host」、「local host」、「hostel」、「ghosts」
*host*
```

- Regexタイプのセクションで指定したファイルの文字列は、PCRE(*Perl互換正規表現*)の通常の拡張機能として解釈されます。文字列と正規表現の照合は、大文字と小文字を区別しないで行われます。正規表現の例:

```
#IPv4
(\d{1,3}.){3}\d{1,3}

#.comドメインのメールアドレス
\w+@\w+\.com
```

- *Cidr*タイプのセクションで指定したファイルの文字列は、IPアドレスまたはIPアドレス範囲として解釈されます。IPアドレスとIPアドレス範囲では、IPv4形式とIPv6形式が許可されます。サブネットマスクは、ビット（オクテット）形式とCIDR（*Classless Inter-Domain Routing*）表記で指定できます。例:

```
#IPv4
192.168.0.1
192.168.0.0/12
192.168.0.0/255.255.255.224

#IPv6
fe80::c7e8/32
fe80::c7e8/255.255.255.224
```

4.Pq、Mysql、Sqliteタイプのセクションで使用されるパラメータ

Conn *{string}*	データベース接続文字列。 **使用可能な値**: • Mysql（**MySQL**）、Pq（**PostgreSQL**）セクションの場合: tcp://[<user>[:<password>]@]<host>[:<port>][/<database name>][?<parameter>=<value>[&...]] unix://[<user>[:<password>]@]<path to socket>[:<database name>][?<parameter>=<value>[&...]] **例**:「tcp://user:pwd@localhost:1234/userdb」、「unix://user:pwd@/tmp/pgsql.sock:userdb」 URI要件に注意してください。 • Sqlite（**SQLite**）セクションの場合: データベースファイルへのパス（file://スキームプレフィックスを指定）。 **例**:「file:///home/user/users.db」 **デフォルト値**:*対応する***DefaultConn**パラメータ値によって定義
Request *{string}*	データベースへのSQLクエリー文字列（SELECT）。ADおよびLDAPタイプのソースについては、以下の自動的に許可されるマーカーをクエリーで使用できます。 • $u、$Uは、クライアントコンポーネントによって送信されたユーザー名に自動的に置き換えられます。 • $d、$Dは、クライアントコンポーネントによって送信されたドメインに自動的に置き換えられます。 • $$は、「$」文字に置き換えられます。 **例**:"SELECT username FROM users INNER JOIN domains ON users.domain = domains.id WHERE domains.name = $d AND users.name = $u" **デフォルト値**:*（未設定）*

注意

SQLクエリーとして、SELECTタイプのクエリーのみ指定できます。置換を実行した後、クエリーは「そのまま」データベースに送信されます。クエリー結果に複数のカラムが含まれる場合、最初のカラムを除くすべてのカラムが無視されます。

5.Redisタイプのセクションで使用されるパラメータ

Conn *{string}*	**Redis**データストレージとの接続文字列。 **使用可能な値**: • tcp://[*<password>*@]*<host>*[:*<port>*][/*<database index>*] • unix://[*<password>*@]*<socket path>*[:*<database index>*] URI要件に注意してください。 **例**:「tcp://localhost:6379」 **デフォルト値**:**RedisDefaultConn**パラメータ値によって定義
Request *{string}*	**Redis**ストレージのクエリー文字列。クエリーでは、以下の自動的に許可されるマーカーを使用できます。 • $u、$Uは、クライアントコンポーネントによって送信されたユーザー名に自動的に置き換えられます。 • $d、$Dは、クライアントコンポーネントによって送信されたドメインに自動的に置き換えられます。 • $$は、「$」文字に置き換えられます。 **例**:"HVALS bad_users" **デフォルト値**:*(未設定)*

*注意:*クエリー結果に複数のカラムが含まれる場合、最初のカラムを除くすべてのカラムが無視されます。

新しいデータソース用のセクションを追加する

Dr.Web for UNIX Internet Gateways用のコマンドラインベースの管理ツールであるDr.Web Ctl(drweb-ctlコマンド経由でアクセス)を使用し、*<名前>*タグを使って、サポートされるタイプの新しいデータソース用の新しいセクションを追加するには、次のコマンドを使用する必要があります。

```
# drweb-ctl cfset LookupD.<type> -a <name>
```

例:

```
# drweb-ctl cfset LookupD.AD -a WinAD1
# drweb-ctl cfset LookupD.AD.WinAD1.Host 192.168.0.20
```

最初のコマンドは[LookupD.AD.WinAD1]という名前のセクションを設定ファイルに追加し、2番目のコマンドはこのセクション内の**ホスト**パラメータの値を変更します。

あるいは、次のように新しいセクションを設定ファイルに直接書き込む（ファイルの末尾に追加するなど）こともできます。

```
[LookupD.AD.WinAD1]
Host = 192.168.0.20
```

どちらの方法でも効果は同じですが、設定ファイルを編集する場合は、**drweb-configd**コンポーネントにSIGHUP信号を送信して変更した設定を適用する必要があります（これを行うには**drweb-ctl**の`reload`コマンドを発行します）。

Dr.Web StatD

Dr.Web StatDコンポーネントは、Dr.Web for UNIX Internet Gatewaysコンポーネントの操作中に発生するイベントの統計を蓄積するために設計されています。イベントは無期限のリポジトリに保存され、リクエストに応じて取得できます。

動作原理

このコンポーネントは、Dr.Web for UNIX Internet Gatewaysコンポーネントの操作中に取得されたイベントの蓄積と無期限の保存を確保します。次のタイプのイベントがログに記録されます。

- コンポーネントの緊急シャットダウン。
- 脅威の検出（特に電子メールメッセージ）

Dr.Web StatDはデーモンモードで動作し、設定制御デーモンによって自動的に起動されます。イベントの表示と管理は、Dr.Web Ctlユーティリティのcommandイベントによって徹底されます。

コマンドライン引数

Dr.Web StatDを実行するには、コマンドラインに次のコマンドを入力します。

```
$ <opt_dir>/bin/drweb-statd [<parameters>]
```

Dr.Web StatDは次のオプションを処理できます。

パラメータ	説明
`--help`	**機能**：コマンドラインパラメータに関する簡単なヘルプ情報をコンソールまたはターミナルエミュレーターに出力し、完了時に終了するように指示します。 **短縮形**： `-h` **引数**： None
`--version`	**機能**：このコンポーネントのバージョンに関する情報をコンソールまたはターミナルエミュレーターに出力し、完了後に終了するように指示します。 **短縮形**： `-v` **引数**： None

例：

```
$ /opt/drweb.com/bin/drweb-statd --help
```

このコマンドはDr.Web StatDに関する簡単なヘルプ情報を出力します。

スタートアップノート

コンポーネントは、自律モードで（他のコンポーネントから自律的に）OSのコマンドラインから直接起動することはできません。必要に応じて、Dr.Web ConfigD設定デーモンによって自動的に起動されます。コンポーネントの

動作を管理するには、Dr.Web for UNIX Internet Gateways用のDr.Web Ctlコマンドラインベースの管理ツールを使用できます（これは**drweb-ctl**コマンドを使用して呼び出されます）。

コマンドラインから製品のこのコンポーネントに関するドキュメントを要求するには、次のコマンドを使用します。**man** 1 drweb-statd

設定パラメータ

コンポーネントは、Dr.Web for UNIX Internet Gatewaysの統合された設定ファイルの[StatD]セクションで指定されている設定パラメータを使用します。

セクションには以下のパラメータが含まれています。

LogLevel *{logging level}*	コンポーネントのログの詳細レベル パラメータ値が指定されていない場合は、[Root]セクションの**DefaultLogLevel**パラメータの値が使用されます。 **デフォルト値**：Notice
Log *{log type}*	コンポーネントのロギング方式。 **デフォルト値**：Auto
ExePath *{path to file}*	コンポーネントの実行ファイルへのパス。 **デフォルト値**：*<opt_dir>*/bin/drweb-statd • **GNU/Linux**の場合：/opt/drweb.com/bin/drweb-statd • **FreeBSD**の場合：/usr/local/libexec/drweb.com/bin/drweb-statd
RunAsUser *{UID / user name}*	このパラメータは、コンポーネントを実行するユーザー名を決定します。ユーザー名は、ユーザーのUIDまたはユーザーのログインとして指定できます。ユーザー名が数字で構成されている場合（UIDに似ている場合）は、「name:」というプレフィックスを付けて指定します。次に例を示します。**RunAsUser** = name:123456。 *ユーザー名が指定されていない場合、コンポーネント操作は開始後にエラーが発生して終了します。* **デフォルト値**：drweb
IdleTimeLimit *{time interval}*	コンポーネントの最大アイドル時間。指定された時間が経過すると、コンポーネントはシャットダウンします。 最小値 - 10s. **デフォルト値**：30s
MaxEventStoreSize *{size}*	イベントデータベースの最大許容サイズ。mbで定義されます。例：**MaxEventStoreSize** = 100mb。 最小値 - 50mb。

	デフォルト値：1GB

付録

付録 A. コンピューター脅威の種類

本マニュアルでは、コンピューターやネットワークに対して潜在的または直接的な損害を与え、ユーザーの情報や権限を侵害するあらゆる種類のソフトウェアを*「脅威」*と定義します（悪意のあるソフトウェアやその他の不要なソフトウェア）。広義では、コンピューターまたはネットワークのセキュリティに対するあらゆる種類の潜在的な危険（すなわちハッカー攻撃につながる脆弱性）を指して「脅威」とする場合があります。

以下に記載するすべての種類のプログラムは、ユーザーのデータまたは機密情報を危険にさらすものです。姿を隠さないプログラム（スパム配信ソフトウェアやさまざまなトラフィックアナライザなど）は、状況によっては脅威と化す可能性はありますが、通常はコンピューター脅威と見なされません。

コンピューターウイルス

この種類のコンピューター脅威は、他のオブジェクト内にそのコードを埋め込む（これを感染と呼びます）ことができるという特徴を持っています。多くの場合、感染したファイルはそれ自体がウイルスのキャリアとなり、また埋め込まれたコードは必ずしもオリジナルのものと一致するとは限りません。ほとんどのウイルスは、システム内のデータを破損させる、または破壊する目的を持っています。

Doctor Webの分類では、ウイルスは感染させるオブジェクトの種類に応じて分けられます：

- *ファイルウイルス* - OSのファイル（通常、実行ファイルおよびダイナミックライブラリ）を感染させ、そのファイルの起動と同時にアクティブになります。
- *マクロウイルス*は、**Microsoft®Office**やマクロコマンドをサポートする他のアプリケーション（Visual Basicで書かれたものなど）が使用するドキュメントに感染するウイルスです。*マクロコマンド*は、完全に機能するプログラミング言語で書かれた一種の実装プログラム（マクロ）です。たとえば、**Microsoft® Word**では、文書を開くか、閉じるか、保存すると、マクロが自動的に開始されることがあります。
- *スクリプトウイルス*はスクリプト言語を使用して作成され、他のスクリプト（OSのサービスファイルなど）に感染します。これらはスクリプトを実行できる他のファイル形式に感染することができるため、Webアプリケーションのスクリプトの脆弱性を利用します。
- *ブートウイルス*は、ディスクやパーティションのブートレコード、またはハードドライブのマスターブートレコードに感染します。多くのメモリを必要とせず、システムのロールアウト、再起動、またはシャットダウンが実行されるまでタスクを実行し続けられます。

多くのウイルスは検出に対抗する何らかの手段を持ち、その手法は常時改良され続けています。すべてのウイルスは、その使用する手法に応じて分類できます。

- *暗号化ウイルス*は、感染するたびにコードを暗号化して、ファイル、ブートセクター、またはメモリ内での検出を妨げます。このようなウイルスのすべてのコピーには、ウイルスのシグネチャとして使用される可能性がある小さな共通コードの一部（復号手順）しか含まれていません。
- *ポリモーフィック型ウイルス*も同様に自身のコードを暗号化しますが、コピーごとに異なる特別な復号プロシージャの生成も行います。つまり、この種のウイルスにはシグネチャバイトがありません。
- *ステルスウイルス*は、特定のアクションを実行して、感染したオブジェクトでの活動と存在を隠します。このようなウイルスは、オブジェクトを感染させる前にそのオブジェクトの特性を収集し、Scannerが変更されたファイルを探し出す際に誤認させるための「ダミー」特性を作り出します。

ウイルスは、書かれているプログラミング言語（ほとんどの場合アセンブラ、高級プログラミング言語、スクリプト言語など）、または感染させるOSに応じて分類することもできます。

コンピューターワーム

「コンピューターワーム」型の悪意のあるプログラムは、ウイルスやその他のマルウェアよりも多く見られるようになってきています。ウイルス同様、自身を複製し拡散できますが、他のオブジェクトを感染させることはありません。ネットワークを通じて（通常、メールの添付ファイルとして）侵入し、ネットワーク内にある他のコンピューターにコピーを拡散します。ユーザーの操作に応じて、または攻撃するコンピューターを選択する自動モードで拡散を開始します。

ワームは1つのファイル（ワームの本体）のみで構成されているとは限りません。多くのワームが、メインメモリ（RAM）内に読み込んだ後にワームの本体を実行ファイルとしてネットワーク経由でダウンロードする感染部分（シェルコード）を持っています。シェルコードがシステム内に存在するだけであれば、システムを再起動することで（RAMが削除されリセットされます）ワームを削除できますが、ワームの本体がコンピューターに侵入してしまった場合はアンチウイルスプログラムのみが対処可能です。

ワームはその驚異的な拡散速度によって、ペイロードを持っていない（直接的な被害を与えない）場合であっても、ネットワーク全体の機能を破壊する能力を持っています。

Doctor Webの分類では、ワームはその拡散方法によって以下のように分けられます。

- *ネットワークワーム*は、さまざまなネットワークとファイル共有プロトコル経由で自身のコピーを拡散します。
- *メールワーム*は、メールプロトコル（POP3、SMTPなど）を使用して拡散します。
- *チャットワーム*は、広く使用されているメッセンジャーやチャットプログラム（ICQ、IM、IRCなど）のプロトコルを使用します。

トロイの木馬プログラム

この種類のコンピューター脅威は自身を複製せず、他のプログラムを感染させません。トロイの木馬は頻繁に使用されるプログラムに成り代わり、その機能を実行します（または動作を模倣します）。同時に、システム内で悪意のある動作（データを破損または破壊、機密情報を送信など）を実行したり、ハッカーが許可なしにコンピューターにアクセス（たとえば第三者のコンピューターに損害を与えるために）したりすることを可能にします。

トロイの木馬のマスキングと悪意のある機能は、ウイルスのそれに似ています。トロイの木馬は、ウイルスのコンポーネントである場合もあります。しかし、ほとんどのトロイの木馬は、ユーザーまたはシステムタスクによって起動される個別の実行ファイルとして配布されます（ファイル交換サーバー、リムーバブルストレージ、メール添付ファイルなどを介して）。

トロイの木馬はよくウイルスやワームによって拡散されることや、他の種類の脅威によっても実行されうる悪意のある動作の多くがトロイの木馬にも起因することから、その分類が難しくなっています。以下のトロイの木馬は、Doctor Webでは個別のクラスとして分類されています。

- *バックドア*は、侵入者がシステムにログオンしたり、既存のアクセスやセキュリティ対策を回避する特権機能を取得したりすることを可能にするトロイの木馬です。バックドアはファイルに感染しませんが、レジストリキーを変更して自身をレジストリに書き込みます。
- *ルートキット*は、自身を隠すためにOSのシステム機能を監視する際に使用されます。さらに、ルートキットは、他のプログラム（他の脅威）、レジストリキー、フォルダ、ファイルのプロセスを隠すことができます。ルートキットは独立したプログラムとして、または別の悪意のあるプログラムのコンポーネントとして拡散されます。動作モードに応じて2種類のルートキットがあります。ユーザーモードで動作（ユーザーモードライブラリの機能を監視）する

*ユーザーモードのルートキット（UMR）*とカーネルモードで動作する（システムのカーネルレベルで機能を監視して、悪意のあるプログラムを検出しにくくする）*カーメルモードのルートキット（KMR）*。

- *キーロガー*は、ユーザーがキーボードを使って入力したデータを記録するために使用されます。目的は、個人情報（ネットワークパスワード、ログイン、クレジットカードデータなど）を盗むことです。
- *クリッカー*は、Webサイトのトラフィックを増加させる目的で、またはDDoS攻撃を実行するためにハイパーリンクを別の（ときに有害な）アドレスにリダイレクトします。
- *プロキシ型トロイの木馬* - 被害者のコンピューターを介して匿名でインターネットにアクセスします。

トロイの木馬は、Webブラウザのスタートページを変更したり特定のファイルを削除したりするなど、これら以外の悪意のある動作も実行することがあります。ただしそのような動作もまた、他の種類の脅威（ウイルスやワーム）によって実行される場合があります。

ハッキングツール

ハッキングツールは、侵入者によるハッキングを可能にするプログラムです。最も一般的なものは、ファイアウォールまたはその他のコンピューター保護システムコンポーネントの脆弱性を検出するポートスキャナです。それらのツールはハッカーだけではなく、管理者がネットワークのセキュリティを検査するためにも用いられます。ハッキングに使用することのできる一般的なソフトウェアや、ソーシャルエンジニアリングテクニックを使用するさまざまなプログラムもハッキングツールに含まれることがあります。

アドウェア

通常、ユーザーの画面に強制的に広告を表示させるフリーウェアプログラム内に組み込まれたプログラムコードを指します。ただしそのようなコードは、他の悪意のあるプログラム経由で配布されてWebブラウザ上に広告を表示させる場合もあります。アドウェアプログラムの多くは、スパイウェアによって収集されたデータを用いています。

ジョークプログラム

アドウェア同様、この種類の脅威はシステムに対して直接的な被害を与えることはありません。ジョークプログラムは通常、実際には起こっていないエラーに関するメッセージを表示させ、データの損失につながるアクションの実行を要求します。その目的はユーザーを驚かせ不快感を与えることにあります。

ダイアラー

幅広く電話番号をスキャンし、モデムとして応答するものを見つけるための特別なコンピュータープログラムです。その後、攻撃者がその番号を使用することによって被害者に通話料の請求書が送られます。または被害者が気づかぬうちに、モデム経由で高額な電話サービスに接続されます。

リスクウェア

これらのソフトウェアアプリケーションは悪意のある目的のために作成されたものではありませんが、コンピューターセキュリティに対する脅威となりうる特徴を持っているため、危険度の低い脅威として分類されます。リスクウェアプログラムはデータを破損または削除してしまう可能性があるのみならず、クラッカー（悪意のあるハッカー）や悪意のあるプログラムによって、システムに被害を与える目的で使用されることがあります。そのようなプログラムの中には、さまざまなリモートチャットおよび管理ツール、FTPサーバーなどがあります。

疑わしいオブジェクト

ヒューリスティックアナライザによって検出される潜在的なコンピューター脅威です。これらのオブジェクトは、あらゆるタイプの脅威（ITセキュリティの専門家も把握していないもの）である可能性があり、または誤検出の場合もあります。疑わしいオブジェクトを含むファイルを隔離に移動するよう選択することをお勧めします。解析のためにDoctor Webアンチウイルスラボにも送信する必要があります。

付録 B. コンピューター脅威の駆除

Doctor Webアンチウイルスソリューションは、悪意のあるソフトウェア検出に複数の手法を同時に使用します。それにより、感染が疑われるファイルに対する徹底的なスキャンを実行し、ソフトウェアの動作を管理できます。

検出方法

シグネチャ解析

スキャンはまず、ファイルコードセグメントを既知のウイルス署名と比較するシグネチャ解析で始まります。シグネチャはウイルスを特定するために必要かつ十分な、連続するバイトの有限なシーケンスです。シグネチャ辞書のサイズを抑えるため、Dr.Webアンチウイルスソリューションはシグネチャのシーケンス全体ではなくチェックサムを使用します。チェックサムはシグネチャを特定し、ウイルス検出および駆除の正確さを維持します。Dr.Webウイルスデータベースは、いくつかのエントリによって、特定のウイルスのみでなく脅威のクラス全体を検出できるよう設計されています。

Origins Tracing™

シグネチャ解析の完了後、Dr.Webアンチウイルスソリューションは既知の感染メカニズムを用いる新種・亜種ウイルスを検出するため、ユニークなテクノロジーOrigins Tracing™を使用します。それにより、Dr.Webユーザーはランサムウェアである**Trojan.Encoder.18**（別名**gpcode**）のような悪質な脅威から保護されます。新種・亜種ウイルスの検出を可能にする他、Origins Tracing™はDr.Webヒューリスティックアナライザによる誤検出を劇的に減らします。Origins Tracing™アルゴリズムを使用して検出されたオブジェクトの名前には`.Origin`拡張子が付きます。

実行のエミュレーション

プログラムコードエミュレーションの技術は、チェックサムによる検索が直接適用できない場合、または実行するのが非常に困難な場合（安全な署名を構築することが不可能なため）に、ポリモーフィック型ウイルスと暗号化ウイルスの検出に使用されます。この方法は、*エミュレーター*、つまりプロセッサーとランタイム環境のプログラミングモデルによる解析コード実行のシミュレーションを意味します。エミュレーターは保護されたメモリ領域（*エミュレーションバッファ*）で動作し、解析されたプログラムの実行は命令ごとにモデル化されます。ただし、これらの命令は実際にはCPUによって実行されるものではありません。エミュレーターがポリモーフィック型ウイルスに感染したファイルを受信すると、エミュレーションの結果は復号されたウイルスコードになります。これは、シグネチャチェックサムを検索することで簡単に判別できます。

ヒューリスティック解析

ヒューリスティックアナライザの検出手法は、ウイルスコードに典型的な、または非常にまれな特徴（属性）に関する特定の情報に基づいています（*ヒューリスティック*）。各属性は、その深刻度および信頼度を定義する*重み*係数を持っています。属性が悪意のあるコードであることを示している場合には重み係数がプラスになり、コンピューター脅威の特徴を示していない場合はマイナスになります。ヒューリスティックアナライザはファイルの重み付け合計値に応じて、未知のウイルスに感染している可能性を計算します。それらの合計が一定のしきい値を超えて

いる場合、ヒューリスティックアナライザによって、オブジェクトは未知のウイルスに感染している可能性があると判定されます。

ヒューリスティックアナライザはファイル解凍の柔軟なアルゴリズムである FLY-CODE™テクノロジーも使用します。このテクノロジーは、Dr.Webにとって既知のパッカーのみでなく、これまでに発見されていない未知のパッカーによって圧縮されたファイル内に悪意のあるオブジェクトが存在する可能性をヒューリスティックに検出します。Dr.Webアンチウイルスソリューションはパックされたオブジェクトのスキャン中に構造エントロピー解析も使用します。このテクノロジーはコードの配置を解析することで脅威を検出します。そのため、1つの検体から、同じポリモーフィックパッカーによってパックされた他の多くの脅威を検出することが可能になります。

不確実な状況で仮説を扱うあらゆるシステム同様、ヒューリスティックアナライザもまたタイプ Iまたはタイプ IIのエラーを生じさせる可能性があります（ウイルスを見逃す、または誤検知）。そのため、ヒューリスティックアナライザによって検出されたオブジェクトは「疑わしい」オブジェクトとして定義されます。

上のスキャン手法に加え、Dr.Webアンチウイルスソリューションは既知の悪意のあるソフトウェアに関する最も新しい情報も使用します。Doctor Webアンチウイルスラボ のエキスパートによって新しい脅威が発見されると、そのウイルスシグネチャ、振る舞い特性、属性を追加した更新が即座に配信されます。更新は1時間に数回行われる場合もあり、たとえ新種のウイルスがDr.Web常駐保護を通過してシステムに侵入した場合でも、更新後には検出され駆除されます。

クラウドベースの脅威検出テクノロジー

クラウドベースの検出方法では、あらゆるオブジェクト（ファイル、アプリケーション、ブラウザ拡張機能など）をハッシュ値によってスキャンします。ハッシュは、特定の長さの数字と文字からなる一意のシーケンスです。ハッシュ値による分析では、オブジェクトは既存のデータベースを使用してスキャンされ、カテゴリー別に分類されます（クリーン、疑わしい、悪意のある、など）。

このテクノロジーにより、ファイルスキャンの時間を最適化し、デバイスリソースを節約することができます。分析されるのはオブジェクトではなく、その固有のハッシュ値であるため、オブジェクトが悪意のあるものであるかどうかの決定はほとんど瞬時に行われます。Dr.Web Cloudサーバーに接続されていない場合、ファイルはローカルでスキャンされ、接続が復元されるとクラウドスキャンが再開されます。

Dr.Web Cloudサービスは多くのユーザーから情報を収集し、これまで未知であった脅威に関するデータを迅速に更新します。これにより、デバイス保護の効果を高めます。

アクション

コンピューター脅威を回避するために、Dr.Web製品は悪意のあるオブジェクトに対してさまざまなアクションを適用します。ユーザーはデフォルト設定を使用、自動的に適用するアクションを設定、あるいは検出のたびに手動でアクションを選択できます。使用可能なアクションは以下のとおりです。

- **Ignore** *(無視)* - いずれのアクションも実行せず、検出された脅威をスキップするように指示します。
- **Report** *(報告する)* - 他のすべてのアクションを実行せずに、検出された脅威について通知します。
- **Cure** *(修復)* - 感染したオブジェクトから悪意のあるコンテンツのみを削除し、修復します。ただし、すべての種類の脅威に対して適用できるわけではありません。
- **Quarantine** *(隔離)* - 検出された脅威を特別なフォルダに移し、残りのシステムから隔離します。
- **Delete** *(削除)* - 感染したオブジェクトを永久に削除します。

コンテナ（アーカイブ、メール添付ファイルなど）内のファイルで脅威が検出された場合は、削除アクションの代わりにコンテナの隔離への移動が実行されます。

付録 C. テクニカルサポート

Dr.Web製品のインストールまたは使用中に問題が発生した場合、テクニカルサポートへのお問い合わせの前に以下のオプションをご利用ください:

- https://download.drweb.com/doc/ から最新のマニュアルやガイドをダウンロードして読む。
- https://support.drweb.com/show_faq/ で「よくあるご質問」を読む。
- https://forum.drweb.com/ でDr.Webフォーラムを見る。

問題が解決しなかった場合、サポートサイト https://support.drweb.com/ の該当するセクション内でwebフォームに必要事項を入力し、直接 Doctor Web テクニカルサポートまでお問い合わせください。

企業情報については、Doctor Web 公式サイト https://company.drweb.com/contacts/offices/ をご覧ください。

問題に対する円滑な対応を可能にするため、テクニカルサポートにご連絡いただく前に、インストールされた製品とその設定、およびシステム環境に関するデータセットを生成することをお勧めします。これは、Dr.Web for UNIX Internet Gatewaysディストリビューションに含まれている特別なユーティリティを使用して行うことができます。

テクニカルサポートに提出するデータを収集するには、次のコマンドを使用します。

```
# <opt_dir>/bin/support-report.sh
```

<opt_dir> は、実行可能ファイルとライブラリを含むDr.Web for UNIX Internet Gatewaysファイルのディレクトリです（**GNU/Linux** の場合はデフォルトで `/opt/drweb.com`）。ディレクトリに使用される規則については、はじめに を参照してください。

テクニカルサポートに提出するデータを収集する際は、ユーティリティをスーパーユーザー権限（*root* ユーザーの権限）で起動することをお勧めします。権限を昇格するには、**su** コマンド（カレントユーザーを変更する）または **sudo** コマンド（指定されたコマンドを別のユーザーの権限で実行する）を使用します。

ユーティリティは次の情報を収集してアーカイブします。

- OSに関するデータ（名前、アーキテクチャ、**uname** `-a` コマンドの結果）
- Doctor Webパッケージを含む、システムにインストールされているパッケージのリスト
- ログの内容
  - Dr.Web for UNIX Internet Gatewaysのログ（コンポーネントごとに設定されている場合）
  - **syslog** システムデーモンのログ（`/var/log/syslog`, `/var/log/messages`）
  - システムパッケージマネージャーのログ（**apt**、**yum** など）
  - **dmesg** のログ
- 次のコマンドの出力：**df**、**ip** `a`（**ifconfig** `-a`）、**ldconfig** `-p`、**iptables-save**、**nft** `export xml`
- Dr.Web for UNIX Internet Gatewaysの設定と構成に関する情報：

- ダウンロードされたウイルスデータベースのリスト（**drweb-ctl** `baseinfo -l`）
- Dr.Web for UNIX Internet Gatewaysディレクトリにあるファイルのリストとそれらファイルの MD5ハッシュ値
- Dr.Web Virus-Finding Engineスキャンエンジンのバージョンと MD5ハッシュ値
- Dr.Web for UNIX Internet Gatewaysの設定パラメーター（`drweb.ini` の内容、ルール、ルールで使用される値のファイル、Luaプロシージャなどを含む）
- Dr.Web for UNIX Internet Gatewaysがスタンドアロンモードで動作している場合、キーファイルから取得したユーザーの情報と権限

製品とそのシステム環境に関する情報を含むアーカイブは、ユーティリティを起動したユーザーのホームディレクトリに保存されます。ファイルの名前は次のようになります。

```
drweb.report.<timestamp>.tgz
```

<timestamp> は、レポート作成の完全なタイムスタンプ（ミリ秒単位）です（例：`20190618151718.23625`）。

付録D. Dr.Web for UNIX Internet Gateways設定ファイル

すべてのDr.Web for UNIX Internet Gatewaysコンポーネントの設定パラメータは、特別な調整デーモンDr.Web ConfigDによって管理されます。これらのパラメータは`drweb.ini`ファイルに格納されています。デフォルトディレクトリは*<etc_dir>*です（**GNU/Linux**の場合は`/etc/opt/drweb.com`）。

設定ファイルには、デフォルト値と異なるパラメータのみが格納されています。パラメータが設定ファイルにない場合は、そのデフォルト値が使用されます。

<opt_dir>、*<etc_dir>*、*<var_dir>*の表記規則の詳細は、はじめにを参照してください。

次のコマンドを使用して、設定ファイルにないパラメータとデフォルト値を持つパラメータを含む、すべての設定のリストを表示できます。

```
$ drweb-ctl cfshow
```

パラメータ値は次のいずれかの方法で変更できます。

1. 設定ファイルでパラメータを指定（任意のテキストエディターでファイルを編集）し、変更を適用するためのSIGHUPシグナルを設定デーモン（**drweb-configd**コンポーネント）に送信します（これを行うには**drweb-ctl** `reload`コマンドを発行します）。
2. このコマンドをコマンドラインに入力します。

```
# drweb-ctl cfset <section>.<parameter> <new value>
```

このコマンドは、管理ツールDr.Web Ctlがスーパーユーザー権限で実行されている場合にのみ実行できます。スーパーユーザー権限を取得するには、**su**コマンドか**sudo**コマンドを使用します。

コマンドライン管理ツールDr.Web Ctl（**drweb-ctl**モジュール）の`cfshow`と`cfset`のコマンド構文の詳細については、Dr.Web Ctlのセクションを参照してください。

ファイル構造

設定ファイルの構造は以下のとおりです。

- ファイルの内容は名前が付けられたセクションに分割されます。こうしたセクションで利用可能な名前は厳密に事前定義されており、変更できません。セクション名は角括弧で指定され、セクションパラメータを使用するDr.Web for UNIX Internet Gatewaysコンポーネント名と似ています（設定デーモンDr.Web ConfigDのすべてのパラメータを保存する[Root]セクションは除く）。
- 設定ファイル内の「;」または「#」文字は、コメントの始まりを示します。文字に続くすべてのテキストは、設定パラメータの読み取り中にDr.Web for UNIX Internet Gatewaysコンポーネントによってスキップされます。
- ファイルの1行には、パラメータ値を1つだけ含めることができます。値を指定する一般的な形式は次のとおりです（「=」文字の前後の空白は無視されます）。

```
<Parameter name> = <Value>
```

- すべてのパラメータ名は厳密に事前定義されており、変更はできません。
- セクション名とパラメータ名はすべて大文字と小文字が区別されません。パラメータ値は、パスに含まれるディレクトリやファイルの名前を除くと、(**UNIX**のようなOSの場合でも)大文字と小文字が区別されません。
- ファイルで指定されるセクションの順序とセクションで指定されるパラメータの順序は重要ではありません。
- 設定ファイルのパラメータ値は引用符で囲むことができ、空白がある場合は引用符で囲む必要があります。
- 一部パラメータは値のリストを持つことができます。その場合、値はコンマで区切るか、設定ファイルの異なる行に複数指定します。前者の場合、コンマの前後の空白は無視されます。空白文字がパラメータ値の一部である場合は、その文字を引用符で囲む必要があります。

  1つのパラメータに複数の値を指定する方法の例:

  1) コンマ区切りリストとして:

  ```
  Parameter = Value1, Value2,"Value 3"
  ```

  2) 設定ファイルの複数行で:

  ```
  Parameter = Value2
  Parameter = Value1
  Parameter = "Value 3"
  ```

  パラメータ値を指定する順序は重要ではありません。

  

  パラメータ値がパスである場合、コンマを使用したコンポーネントのリストが使用されている場合は、一覧表示された各パラメータ値を引用符で囲む必要があります。たとえば、パラメータ**ExcludedPaths**に2つのパス/etc/file1と/etc/file2が必要な場合、このパラメータは1つの文字列として以下のように設定ファイルに書き込むか

  ```
  ExcludedPaths = "/etc/file1", "/etc/file2"
  ```

  または2つの文字列として書き込みます

  ```
  ExcludedPaths = /etc/file1
  ExcludedPaths = /etc/file2
  ```

  上記以外の場合、このパラメータを使用するコンポーネントは、文字列「/etc/file1、/etc/file2」を1つのパスとして解釈します。

- パラメータに複数の値を指定できる場合は、明確に指定されます。そのため、現在のマニュアルや設定ファイルのコメント内で明確に指定されていない場合、パラメータに指定できる値は1つのみです。

設定ファイルのセクションの説明については、Dr.Web for UNIX Internet Gatewaysコンポーネントの説明を参照してください。

パラメータタイプ

設定パラメータは以下のいずれかのタイプになります。

- *address* - *<IP address>:<port>*の値ペアとして指定したネットワーク接続アドレス。場合によっては、ポート値を省略できます（省略した場合は、パラメータの説明で指定されます）。
- *ブール値* - インジケーターとして使用されるフラグ。このようなパラメータは、値が`Yes`または`No`になります。
- *整数* - パラメータ値は、非負の整数になります。
- *小数* - パラメータ値は、小数部分のある非負の整数になります。
- *時間間隔* - パラメータ値は、非負の整数と時間単位を示すサフィックス（文字）で構成される時間間隔になります。以下のサフィックスを使用できます。
  - `w` - 週（`1w = 7d`）
  - `d` - 日（`1d = 24h`）
  - `h` - 時間（`1h = 60m`）
  - `m` - 分（`1m = 60s`）
  - `s` – 秒

  サフィックスを省略した場合、間隔は秒単位と見なされます。秒単位で表される時間間隔では、小数点の後にミリ秒を指定できます（区切り文字の後に3桁以内、0.5秒～500ミリ秒など）。異なる時間単位で複数の時間間隔を指定できます。この場合、間隔の合計が計算されます（実際には、値が設定に書き込まれる前に時間間隔は常にミリ秒に変換されます）。

  基本的には、すべての時間間隔は次の形式で表現されます。N_1`w`N_2`d`N_3`h`N_4`m`N_5[`.`N_6]`s`。N_1, ..., N_6はこの間隔に含まれる時間単位の数です。たとえば、1年（365日）は次のように表すことができます（すべてのレコードは同じ365日を表しています）：`365d`、`52w1d`、`52w24h`、`51w7d24h`、`51w7d23h60m`、`8760h`、`525600m`、`31536000s`。

  以下の例では、30分、2秒、500ミリ秒の間隔を指定する方法を示しています。

  1.設定ファイルでの設定。

  ```
  UpdateInterval = 30m2.5s
  ```

  2.コマンド**drweb-ctl** `cfset`を使用：

  ```
  # drweb-ctl cfset Update.UpdateInterval 1802.5s
  ```

  3.コマンドラインパラメータ経由（例：コマンドライン引数の場合）。

  ```
  $ drweb-se --WatchdogInterval 1802.5
  ```

- *サイズ* - パラメータ値は、オブジェクト（ファイル、バッファ、キャッシュなど）のサイズで表すことができ、非負の整数と単位を表すサフィックスで構成します。以下のサフィックスを使用できます。
  - `mb` - メガバイト（`1mb = 1024kb`）。
  - `kb`—キロバイト（`1kb = 1024b`）。
  - `b` - バイト。

  サフィックスを省略した場合、サイズはバイト単位と見なされます。異なる単位で複数のサイズを指定できます。この場合、サイズの合計が計算されます（実際には、サイズ値は常にバイトに変換されます）。

- *ディレクトリ(ファイル)へのパス* - パラメータ値は、ディレクトリ(ファイル)へのパスである文字列になります。ファイルパスはファイル名で終わる必要があります。

  

  UNIX系システムでは、ディレクトリとファイルの名前は大文字と小文字が区別されます。パラメータ記述で明示的に指定されていない場合、パスに特殊文字(`?`、`*`)を含むマスクを含めることはできません。

- *ロギングレベル* - Dr.Web for UNIX Internet Gatewaysコンポーネントのイベントを記録するレベル。このタイプのパラメータでは、以下の値を使用できます。
  - `DEBUG` - 最も詳細なログレベル。すべてのメッセージとデバッグ情報が登録されます。
  - `INFO` - すべてのメッセージが登録されます。
  - `NOTICE` - すべてのエラーメッセージ、警告、通知が登録されます。
  - `WARNING` - すべてのエラーメッセージと警告が登録されます。
  - `ERROR` - エラーメッセージのみが登録されます。
- *ログタイプ* - パラメータ値でDr.Web for UNIX Internet Gatewaysコンポーネントによるログの実行方法(ロギング方式)を定義します。このタイプのパラメータでは、以下の値を使用できます。
  - `Stderr[:ShowTimestamp]` - メッセージは*stderr*(標準エラーストリーム)に表示されます。この値は設定デーモンの設定で*のみ*使用できます。バックグラウンドモードで動作する(「*デーモン化される*」)場合、つまりパラメータ`-d`を指定して起動した場合、バックグラウンドモードで動作するコンポーネントは端末のI/Oストリームにアクセスできないため、この値は*使用できません*。追加パラメータ`ShowTimestamp`は、すべてのメッセージにタイムスタンプを追加するように指示します。
  - `Auto` -ログ対象のメッセージは設定デーモンDr.Web ConfigDに送られ、設定に基づいて一か所(`[Root]`セクションの**ログ**パラメータ)に保存されます。この値は、*設定デーモンを除く*すべてのコンポーネントに指定され、デフォルト値として使用されます。
  - `Syslog[:<facility>]` - メッセージはシステムロギングサービス**syslog**に送信されます。
  - 追加オプション*<facility>*は、**syslog**のメッセージ登録レベルを指定するために使用します。次の値を使用できます。
    - `DAEMON` - デーモンのメッセージ。
    - `USER` - ユーザープロセスのメッセージ。
    - `MAIL` - メールプログラムのメッセージ。
    - `LOCAL0` - ローカルプロセス0のメッセージ。

    ...
    - `LOCAL7` - ローカルプロセス7のメッセージ。
  - *<path>* - メッセージは指定されたログに直接保存されます。

  パラメータ値の指定方法の例。

  1.設定ファイルでの設定。

  ```
  Log = Stderr:ShowTimestamp
  ```

  2.コマンド**drweb-ctl** `cfset`を使用:

  ```
  # drweb-ctl cfset Root.Log /var/opt/drweb.com/log/general.log
  ```

  3.コマンドラインパラメータ経由(例:コマンドライン引数の場合)。

```
$ drweb-se --Log Syslog:DAEMON
```

- *アクション* - 特定の脅威または別のイベントが検出されたときにDr.Web for UNIX Internet Gatewaysコンポーネントによって実行されるアクション。次の値を使用できます。
  - `Report` - アクションは適用せず、検出された脅威についての通知のみをするよう指示します。
  - `Cure` - 脅威の駆除を試みる(悪意のあるコンテンツのみを削除する)よう指示します。
  - `Quarantine`(隔離) - 感染したファイルを隔離に移動するよう指示します。
  - `Delete` - 感染したファイルを削除するよう指示します。

一部のアクションは特定のイベントにのみ適用できます(たとえば「スキャンエラー」イベントでは`Cure`アクションをトリガーできません)。許可されたアクションは、常に*アクション*タイプのパラメータに記述されます。

他のパラメータタイプと可能な値は、パラメータの説明で指定されています。

トラフィックモニタリングのルール

このセクションの内容:

- 概要
- ルールフォーマット。
- 条件。
- アクション。
- ルールで使用される変数。
- 不要なWebサイトと脅威のカテゴリー。
- ルール条件で使用できる設定パラメータ。
- 設定ファイルにルールを保存する機能。

概要

ルールは、IF *<conditional_part>* THEN *<action_part>*などのプロダクションルールで表されます。*<conditional_part>*の部分には、次のスキャンタイプが指定されます。*「変数値が指定されている(いない)」*または*「変数値が指定セットに含まれている(いない)」*。*<action_part>*には、一連のアクション(1つ以上)が含まれ、各アクションは*最終的な解決*(スキャンしたオブジェクトをスキップまたはブロックする)、または*「スキャンしたオブジェクトの機能を変更」*、*「指定変数にセット値を割り当て」*、または*「指定変数の値の配列へセット値を追加」*などの*修正アクション*になります。

ルールアクションの一部は、条件部分がtrueの場合にのみ実行されます。条件部分がfalseと評価された場合、この規則で指定されたアクションは実行されず、プログラムは次のルールにジャンプします。最終的な解決が実行されるまで、ルールは下に向かって垂直に評価されます。最終的な解決が実行されると、以降のルール(存在する場合)はすべて無視されます。ルールが実行されるとき、*<action_part>*内のアクションが左から右への指定の順序で実行されることが重要です。ルール処理を中断する一連のアクションに最終的な解決策がある場合、*<action_part>*に指定された残りのアクションは実行されません。

ルールフォーマット

ルール作成のフォーマット:

```
[<condition>[, <condition>[, ...]]] : <action>[, <action>[, ...]]
```

ルールの条件付き部分(':'の前)がない場合は、その部分のアクションは条件なしで実行されます。ルールの条件部分がない場合は、':'の区切り文字を省略できます。条件部分の条件とアクション部分のアクションとの間のコンマは、論理積(「and」)の役割を果たします。条件部分はすべての条件がtrueである場合にのみtrueになり、アクション部分に指定されたすべてのアクションは、ルールの処理を中断する最終的な解決まで、左から右への指定の順序で実行されます。ルールでは、キーワード、変数名、設定パラメータにとって登録は重要ではありません。

条件

次のタイプの条件は、ルールの条件部分で使用できます。

条件	条件の意味
<variable> <value >	指定された変数の値が設定値と一致している。 *値の指定が可能な変数にのみ使用できます。*
<variable> `[not] in` *<set of values>*	指定された変数の値が、指定された値のセット<set of values>に含まれている(*notの場合 - 指定された値のセットに含まれない*)。
<variable> `[not] match` *<set of values>*	指定された変数の値は、指定された値のセットにある任意の正規表現と一致する(*notの場合 - 指定された値のセットのどの表現とも一致しない*)。 正規表現は、*POSIX*構文(*BRE*、*ERE*)または*Perl*構文(*PCRE*、*PCRE2*)のいずれかを使用して指定されます。
<variable> `[not] gt` *<value>*	指定された変数の値が設定値以下。 *単一の値を持つことができる変数にのみ使用できます。*
<variable> `[not] lt` *<value>*	指定された変数の値が設定値以上。 *単一の値を持つことができる変数にのみ使用できます。*

*) オプションのキーワードnotは否定を意味します。

変数と比較される*<set of values>*の部分は、以下の方法で指定できます。

構文	意味
(*<value 1>*[, *<value 2>*[, ...]])	括弧内には、チェックする値のセットを直接列挙します(1つ以上の値)。値が1つしかなく、in条件が使用されている場合は、括弧を省略できます(場合によっては*<variable>* *<value>*になります)。
"*<section>*.*<parameter>*"	特定の設定パラメータに現在割り当てられている値のセット。引用符の間には、値(または値のセット)のチェックが必要な設定パラメータの名前を指定します(パラメータが属するセクションの名前も指定する必要があります)。 条件として使用できるパラメータのリストは、ルールが設定されているコンポーネントによって異なります。リストは以下に示します。
file("*<file name>*")	値の一覧はテキストファイル*<file name>*から読み込まれます(1つのファイル文字列 - 1つのリスト要素、文字列の先頭と末尾のスペースは無視されます)。ファイルへのパスは絶対パスである必要があります。*<file name>*に引用符とアポストロフィが含まれている場合は、それらをエスケープ('\')する必要があります。 ファイルサイズは64 MB未満である必要があります。 設定ファイルのダウンロード中に1回、ファイルの内容が読み込まれ、ルールに挿入されます。ファイルがない、またはファイルサイズが超過すると、ダウンロード中にエラーx102が表示されます。 Dr.Web for UNIX Internet Gatewaysの動作中にファイルの内容が変更された場合、変更を適用するには、**drweb-ctl**のreloadコマンドを使用してファイルをリロードする必要があります。 ファイルからの値のセットは、すべての変数に使用できるわけではありません。ファイルからの値のセットを用いて値のスキャンに変数を使用できるかどうかは、以下に示します。

構文	意味
<type_of_LOOKUP_request>@ *<tag>*[@ *<value>*]	値の配列は外部データソースからDr.Web LookupDを介して要求されます。*<LOOKUP_query_type>*はソースのタイプです。*<tag>*は、チェックされたパラメータのサンプリングのための接続を説明するセクションの名前です。オプションの *<value>*は、データソースから抽出された値の配列内に必要な値です。 Dr.Web LookupDの値は、すべての変数に使用できるわけではありません。また、条件の *<scanning>*を適用できない変数もあります。Dr.Web LookupDを用いて値のスキャンに変数を使用できるかどうかは、以下に示します。

変数が複数の値を持つ場合は、指定された変数 *<variable>*の現在値と指定されたセット *<set of values>* の共通集合が空でない場合に、*<variable>* `in` *<set of values>*の条件はtrueになります。逆の場合は、`not in`がtrueになります。たとえば、`X`が変数で、現在の値として*a*、*b*、*c*が設定されているとします。その場合は、

- `X in (a, b)`は、*a*と*b*の値が両方のセットにあるため、trueになります。
- `X in (a, d, e)`は、*a*の値が両方のセットにあるため、trueになります。
- `X in (d, e)`は、変数の値（*a*、*b*、*c*）が（*d*、*e*）のセットにないため、falseになります。
- `X in ()`は、変数の値の配列が空白でないため、falseになります。
- `X not in ()`は、変数の値の配列が空白でないため、trueになります。
- `X not in (d, e)`は、変数の値（*a*、*b*、*c*）がセット（*d*、*e*）にないため、trueになります。
- `X not in (a, d, e)`は、*a*が両方のセットにあるため、falseになります。

以下の変数の説明では、各変数に値のセットを採用できるかどうかを示しています。

アクション

アクションは、オブジェクトの受け渡しを許可するかどうかを決定する*最終的な解決策*と、基本的なルールの条件をチェックするときに使用できる、*いくつかの変数の値を変更するアクション*に分かれます。

最終的な解決

解決策	説明（意味）
一般的な解決策	
`Pass`	トラフィックをスキップします（接続の作成を許可し、オブジェクトを受信者に送信します）。下方ルール（ある場合）は使用されません。

解決策	説明（意味）
`Block as <reason>`	トラフィックをブロックします（接続の作成をブロックし、オブジェクトを受信者に送信します）。下方ルール（設定されている場合）は使用されません。 ブロックの *<reason>*はログに記録されます。同じ理由が、ユーザーに表示されるブラウザ通知を定義する際にも使用されます。`Block`の *<reason>*には2つの標準の理由を使用できます。 • `BlackList` - ブラックリストに含まれているためデータをブロックします。 • `_match` - ルールの実行をトリガーするカテゴリーに属する脅威がWebリソースまたはファイルに含まれているため、ブロックします（`*_category in (...)`の条件の場合）。`_match`変数には、ブロックされたカテゴリーのリストに一致したものが含まれます。

最終的な解決を処理する機能：

- `Block as BlackList`は、常に*「ブラックリストに含まれる」*として処理されます（この解決策に指定された条件を考慮しません）。
- `Block as _match`は、`_match`が空白でない場合は*「_matchカテゴリーに属する」*ものとして処理されます。
- `Block as _match`は、`_match`が空白の場合は*「ブラックリストに含まれる」*として処理されます（この解決策に指定された条件を考慮しません）。
- すべてのルールが考慮されていて、解決策のあるルールがいずれも実行されない（またはルールに解決策がない）場合、この状況は`Pass`アクションと同じです。

変数の値の変更

変数値を変更するには、次の命令を使用します。

```
SET <variable> = ([<value 1>[, <value 2>[, …]]])
```

大括弧内に何もない場合、変数値のリストは消去されます。値が1つしかない場合は、大括弧を省略して次の構文を使用する必要があります。

```
SET <variable> = <value >
```

ルールで使用される変数

ルールの中で変数を示すときは、記号の登録は考慮されません。複数のアクション名を持つ変数は、スペース用のアンダースコアを使用しても使用しなくても保存できます。したがって、レコード`variable_name`、`VariableName`、`variablename`は同じ変数を表します。このセクションでは、すべての変数はアンダースコアを使用して保存されます（つまり、`variable_name`書き込みオプションが使用されます）。

変数	説明	使用可能	
		条件部分	アクション部分（SET）
protocol	接続で使用されるネットワークプロトコルタイプ。 *この変数は、複数の値を指定することができます。* **使用可能な値**：HTTP、SMTP、IMAP、POP3。 **使用方法**： • 変数値は、SSL/TLSが使用されていない場合、またはSSLのラップ解除が許可されている場合にのみ定義できます。 • Dr.Web ICAPDルールにHTTP以外の値を指定することは意味がありません。Dr.Web ICAPDにはHTTPしか指定できません。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例**： `protocol in (HTTP, SMTP)` `protocol in (POP3)` `protocol in file("/etc/file")`	Yes	No
sni_host	SSL/TLSを介して接続が確立されるSNIホスト（アドレス）。 **使用方法**： • SSLが使用されず、変数の値が定義されていない場合、条件はfalseと評価されます。 • Dr.Web ICAPDルールに使用しても意味がありません（SSLは処理されません。そのため条件は常にfalseと評価されます）。 • 変数値をリストとして記述したファイルを参照することも可能です。 • proc変数と組み合わせて使用できます（以下を参照）。 **例**： `sni_host not in ('vk.com', 'ya.ru')` `sni_host in "LinuxFirewall.BlackList"` `sni_host in file("/etc/file")`	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
sni_category	お使いのコンピューターがSSL/TLSを介して接続しようとしているホストで、ホスト（SNIヘッダーから識別される）が属するカテゴリー（*AdultContent*など）のリスト（Webリソースカテゴリーのデータベースに準ずる）。 *この変数は、複数の値を指定することができます。* **使用方法**： • SSLが使用されず、変数の値が定義されていない場合、条件はfalseと評価されます。 • Dr.Web ICAPDルールに使用しても意味がありません（SSLは処理されません。そのため条件は常にfalseと評価されます）。 • Dr.Web ICAPDで使用されるルールの場合、スキャン結果によりホストが事前に定義されたカテゴリー（「安全な」ホスト）のいずれにも属していなくても、`not in`が使用される条件は*true*になります。Dr.Web Firewall for Linux（SpIDer Gate）のルールでは、この場合の条件は*false*になります。 • Webリソースカテゴリーのデータベースがインストールされていない場合、変数はルールで使用できません（ルールの条件がtrueであるかどうかを確認しようとすると、エラーx112が発生します）。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例**： `sni_category not in (AdultContent, Chats)` `sni_category in "LinuxFirewall.BlockCategory"` `sni_category in (FreeEmail)` `sni_category not in file("/etc/file")`	Yes	No
url	クライアントからリクエストされたURL。指定された文字列または正規表現と比較できます。 **使用方法**： • Dr.Web ICAPDのルールでのみ使用できます。	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	• Dr.Web LookupDを使用してこの変数の値を確認できます。 • 変数値をリストとして記述したファイルを参照することも可能です。 • proc変数と組み合わせて使用できます（以下を参照）。 **例：** `url match ("drweb.com", "example\..*", "aaa\.ru/")` `url match "ICAPD.Adlist"` `url not match LDAP@BadURLs` `url match file("/etc/file")`		
url_host	接続が確立されるURL/ホスト。 **使用方法：** • 変数値は、SSL/TLSが使用されていない場合、またはSSLのラップ解除が許可されている場合にのみ定義できます。 • Dr.Web LookupDを使用してこの変数の値を確認できます。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例：** `url_host in ('vk.com', 'ya.ru')` `url_host not in "ICAPD.Whitelist"` `url_host in LDAP@hosts` `url_host not in file("/etc/file")`	Yes	No
url_category	URL/ホストが属するカテゴリーのリスト。情報はカテゴリーのデータベースまたはDr.Web Cloudの応答に基づきます。 *この変数は、複数の値を指定することができます。* **使用方法：** • 変数値は、SSL/TLSが使用されていない場合、またはSSLのラップ解除が許可されている場合にのみ定義できます。 • Dr.Web MailDとDr.Web ICAPDで使用されるルールの場合、スキャン結果によりURL/ホストが事前に定義されたカテゴリー（「安	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	全な」URL/ホスト）のいずれにも属していなくても、not inが使用される条件はtrueになります。Dr.Web Firewall for Linux（SpIDer Gate）のルールでは、この場合の条件はfalseになります。 • Webリソースカテゴリーのデータベースがインストールされていない場合、変数はルールで使用できません（ルールの条件がtrueであるかどうかを確認しようとすると、エラーx112が発生します）。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例：** `url_category not in (AdultContent, Chats)` `url_category in "LinuxFirewall.BlockCategory"` `url_category in (FreeEmail)` `url_category in file("/etc/file")`		
threat_category	転送されたデータにある、脅威が属するカテゴリーのリスト（ウイルスデータベースからの情報に準ずる）。 *この変数は、複数の値を指定することができます。* **使用方法：** • 変数値は、SSL/TLSが使用されていない場合、またはSSLのラップ解除が許可されている場合にのみ定義できます。 • Dr.Web ICAPDで使用されるルールの場合、スキャン結果によりオブジェクトが事前に定義されたカテゴリー（「安全な」オブジェクト）からの脅威を含んでいなくても、not inが使用される条件はtrueになります。Dr.Web Firewall for Linux（SpIDer Gate）のルールでは、この場合の条件はfalseになります。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例：** `threat_category in "LinuxFirewall.BlockThreat"` `threat_category not in (Joke)`	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	`threat_category in file("/etc/file")`		
`user`	その管理者権限によりトラフィックを送信（または受信）しているプロセスが起動されるユーザー名。 **使用方法**： • Dr.Web ICAPDルールでは、プロキシサーバーで認証されたユーザーの名前が暗示されます（プロキシサーバーが認証をサポートしている場合）。プロキシサーバーがユーザー認証をサポートしていない場合、変数には空の値が入ります。 • Dr.Web LookupDを使用してこの変数の値を確認できます。 • ユーザーが特定のユーザーグループに属しているかどうかを調べる必要がある場合、グループのリストを返すLDAPまたはActive Directoryデータソースを使用して、必要なグループ（ユーザーがそのメンバーであるかどうかを確認するグループ）の名前を指定します。*<type of the source for LookupD>@<source of groups>@<required group>*の形式を使用します。Active Directory（AD@）へのリクエストはグループのリストのみを返すため、これらのリクエストでは *@<required group>* 部分を使用する必要があります。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例**： `user in ('user1', 'user2')` `user in AD@Winusergroups@Admins` `user in LDAP@AllowedUsers` `user not in file("/etc/file")`	Yes	No
`src_ip`	接続を確立しているホストのIPアドレス。 **使用方法**： • Dr.Web LookupDを使用してこの変数の値を確認できます。 • 変数値をリストとして記述したファイルを参照することも可能です。	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	例： `src_ip not in (127.0.0.1, 10.20.30.41, 198.126.10.0/24)` `src_ip in LDAP@AllowedAddresses` `src_ip not in file("/etc/file")`		
`proc`	接続を確立するプロセス（完全な実行可能パス）。 **使用方法**： • Dr.Web ICAPDルールに使用しても意味がありません（コンポーネントにはプロセスに関する情報は含まれません。そのため条件は常にfalseと評価されます）。 • 変数値をリストとして記述したファイルを参照することも可能です。 • `sni_host`、`url`、`dst_address`の各変数と組み合わせて使用できます（以下を参照）。 **例**： `proc in ('/usr/bin/ls')` `proc not in ('/home/user/myapp', '/bin/bin1')` `proc in "LinuxFirewall.ExcludedProc"` `proc in file("/etc/file")`	Yes	No
`direction`	接続におけるトラフィックのタイプ。 **使用可能な値**：`request`（クライアントリクエスト）、`response`（サーバー応答）。 *この変数は複数の値を同時に指定することはできません。*`match`と`in`*の条件は適用できません。* **例**： `direction request` `direction not response`	Yes	No
`divert`	接続の方向。	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	使用可能な値：input（受信 - ローカルホストの外部から作成/開始）、output（送信 - ローカルホストで作成/開始）。 *この変数は複数の値を同時に指定することはできません。matchとinの条件は適用できません。* **例**： `divert input` `divert not output`		
content_type	接続中に転送されるデータのMIMEタイプ。 **使用方法**： • SSL/TLSが使用されていない場合、またはSSLのラップ解除が許可されている場合にのみ定義できます。 • 式「*/*」は、任意のMIMEタイプのデータと、ヘッダーContent-TypeなしのHTTP応答に一致します。 • Dr.Web LookupDを使用してこの変数の値を確認できます。 • 変数値をリストとして記述したファイルを参照することも可能です。 **例**： `content_type in ("multipart/byteranges", "application/octet-stream")` `content_type not in ("text/*", "image/*")` `content_type not in ("audio/*")` `content_type in ("*/*")` `content_type in LDAP@BlockedContent` `content_type not in file("/etc/file")`	Yes	No
(proc, *<variable>*)	プロセスのネットワークアクティビティ。ここで、procは完全なプロセス実行可能パス（上記を参照）で、*<variable>*はアクティビティのタイプを定義し、次のいずれかの値になります。 • sni_host - SSL/TLSを介して接続が確立されるSNIホスト（アドレス）。	Yes	No

変数	説明	使用可能	
		条件部分	アクション部分（SET）
	• `url` - クライアントによってリクエストされたURL（上記を参照）。 • `dst_address` - プロセスが接続を確立するネットワークアドレス（*<IP address>*: *<port>*）。 **使用方法**: • 条件一致（{ *<Proc_reg>*, *<Var_reg>*} [, …]）でのみ使用します。ここで、*<Proc_reg>*は`proc`の正規表現で、*<Var_reg>*は *<variable>*の正規表現です。 • Dr.Web ICAPDルールに使用しても意味がありません（コンポーネントにはプロセスに関する情報は含まれません。そのため条件は常にfalseと評価されます）。 **例**: `(proc, url) match ({"/usr/bin/wget", "www\.ya\.*"}) (proc, dst_address) match ({"/usr/bin/.*", "192\.168\.1\.\d+:12345"})`		
`unwrap_ssl`	SSL/TLSを介して転送されたトラフィックがラップ解除されているかどうか。 **使用可能な値**: `true`、`false`。 **使用方法**: • 変数には常に任意の値があります。命令`SET unwrap_ssl = ()`は不可能です。 • 変数は条件として使用できません。SSLのラップ解除のみ管理する必要があります（こちら側がトリガーしたブロッキングについての通知を含むWebページを表示するためなど）。 • Dr.Web ICAPDルールに使用しても意味がありません（SSLは処理されません。変数の変更はルールのプロセスに影響しません）。 **例**: `SET unwrap_ssl = TRUE set Unwrap_SSL = false`	No	Yes

変数	説明	使用可能	
		条件部分	アクション部分（SET）
http_templates_dir	HTTPリクエストをブロックする際の通知ページテンプレートが保存されているディレクトリへのパス。 パスが「/（スラッシュ）」で始まる場合は絶対パスになります。それ以外の記号で始まる場合は、相対パスになります。後者の場合、**TemplatesDir**パラメータで指定されたディレクトリに相対して指定されます。 **使用方法：** • これはHTTP（S）プロトコルでのみ使用できます。 **例：** `SET http_templates_dir = "/etc/mytemplates"` `set http_templates_dir = "templates_for_my_site"`	No	Yes

不要なWebサイトと脅威のカテゴリー

1. 望ましくないWebサイトのカテゴリー（変数sni_category、url_categoryでの使用）

文字・記号	Webサイトのカテゴリー
InfectionSource	悪意のあるソフトウェアを含むWebサイト（「感染源」）。
NotRecommended	閲覧が推奨されない、不正なWebサイト（「ソーシャルエンジニアリング」を使用しているもの）。
AdultContent	ポルノまたはエロティックなコンテンツ、出会い系サイトなどを含むWebサイト。
Violence	暴力行為を助長するWebサイトや、さまざまな死亡事故などに関するコンテンツを含むWebサイト。
Weapons	武器および爆発物に関するWebサイトや、それらの製造に関する情報を提供しているWebサイト。
Gambling	勝負事、カジノ、オークション、オンラインゲームへのアクセスを提供するWebサイト（賭けサイトを含む）。
Drugs	麻薬の使用、製造または流通を促進するWebサイト。
ObsceneLanguage	猥褻な言葉を含む（タイトル、記事などに）Webサイト。
Chats	テキストメッセージのリアルタイム送信を提供するWebサイト。
Terrorism	攻撃的なプロパガンダ、またはテロ攻撃に関する内容を含むWebサイト。

文字・記号	Webサイトのカテゴリー
FreeEmail	メールボックスの無料登録を提供するWebサイト。
SocialNetworks	さまざまなソーシャルネットワークサービス: 一般、仕事、企業、興味、テーマ別出会い系Webサイト。
DueToCopyrightNotice	一部の著作物(映画、音楽など)の著作権者によって定義されるWebサイト。
OnlineGames	インターネットへの常時接続を使用してゲームへのアクセスを提供するWebサイト。
Anonymizers	ユーザーが個人情報を隠し、ブロックされたWebリソースにアクセスすることを可能にするWebサイト。
CryptocurrencyMiningPool	仮想通貨マイニングのための一般的なサービスへのアクセスを提供するWebサイト。
Jobs	求人検索Webサイト。

変数`sni_category`*と*`url_category`*の値として、ブロッキングを制御するパラメータの名前を使用することもできます(以下参照)。*

2. 脅威のカテゴリー(変数`threat_category`用)

文字・記号	脅威のカテゴリー
KnownVirus	既知の脅威(ウイルス)。
VirusModification	既知の脅威(ウイルス)の亜種。
UnknownVirus	未知の脅威、疑わしいオブジェクト。
Adware	アドウェア。
Dialer	ダイアラー。
Joke	ジョークプログラム。
Riskware	リスクウェア。
Hacktool	ハッキングツール。

変数`threat_category`*の値として、ブロッキングを制御するパラメータの名前を使用することもできます(以下参照)。*

ルール条件で使用できる設定パラメータ

Dr.Web Firewall for Linuxのコンポーネントルールで使用されるパラメータ（プレフィックス`LinuxFirewall`で示されます）。

パラメータ	説明と使用例
`Whitelist`	ホワイトリストには、Dr.WebのURLカテゴリーのデータベースに対象のドメインが含まれている場合でも、アクセスが許可されるドメインのリストが含まれます。 **例：** `sni_host in "LinuxFirewall.Whitelist" : Pass` `url_host not in "LinuxFirewall.Whitelist" : Block as _match`
`Blacklist`	ブラックリストにはドメインのリストが含まれ、そのドメインへのアクセスはユーザー（または管理者）によってブロックされます。 **例：** `sni_host in "LinuxFirewall.Blacklist" :SET Unwrap_SSL = FALSE` `url_host in "LinuxFirewall.Blacklist" : Block as BlackList`
`BlockCategory`	「メタパラメータ」：値が、`[LinuxFirewall]`セクションで対応する**`Block *`**パラメータが`Yes`に設定されているWebリソースカテゴリー（*Chats*、*AdultContent*など）の名前のリスト。 **例：** `url_category in "LinuxFirewall.BlockCategory" : Block as _match` `sni_category in "LinuxFirewall.BlockCategory" : Block as BlackList`
`BlockThreat`	「メタパラメータ」：値が、`[LinuxFirewall]`セクションで対応する**`Block *`**パラメータが`Yes`に設定されている脅威タイプ（*KnownVirus*、*Joke*など）の名前のリスト。 **例：** `threat_category in "LinuxFirewall.BlockThreat" : Block as _match`
`ExcludedProc`	トラフィックをスキャンから除外する必要がある信頼できるプロセスのリスト。 **例：** `proc in "LinuxFirewall.ExcludedProc" : Pass`

Dr.Web ICAPDのコンポーネントルールで使用されるパラメータ（プレフィックスICAPDで示されます）。

パラメータ	説明と使用例
Whitelist	ホワイトリストには、Dr.WebのURLカテゴリーのデータベースに対象のドメインが含まれている場合でも、アクセスが許可されるドメインのリストが含まれます。 **例：** `url_host not in "ICAPD.Whitelist" : Block as BlackList`
Blacklist	ブラックリストにはドメインのリストが含まれ、そのドメインへのアクセスはユーザー（または管理者）によってブロックされます。 **例：** `url_host in "ICAPD.Blacklist" : Block as BlackList`
Adlist	広告リスト。広告Webサイトを説明する正規表現のリストが含まれています。リストはユーザー（または管理者）によって定義されます。 **例：** `url match "ICAPD.Adlist" : Block as BlackList`
BlockCategory	「メタパラメータ」：値が、[ICAPD]セクションで対応する**Block *パラメータがYesに設定されているWebリソースカテゴリー（*Chats*、*AdultContent*など）の名前のリスト。 **例：** `url_category in "ICAPD.BlockCategory" : Block as _match`
BlockThreat	「メタパラメータ」：値が、[ICAPD]セクションで対応する**Block *パラメータがYesに設定されている脅威タイプ（*KnownVirus*、*Joke*など）の名前のリスト。 **例：** `threat_category in "ICAPD.BlockThreat" : Block as _match`

設定ファイルにルールを保存する機能

- 設定ファイルのルールを使用するコンポーネントの設定セクションで、ルールは**RuleSet**などの変数で保存され、それぞれが無制限のルールのセット（シーケンス）となります。さらに、各セットのルールは、最終的に解決されるまで、順次（下に向かって）考慮されます。
- 無条件のルール（条件部分のないアクションのみを含むルール）を設定ファイルに書き込む場合、空の条件部分と区切り文字「：」が追加されます。

  たとえば、次のルールは、条件部分を含まず*アクションのみで構成*されています。

  ```
  Block as _match
  ```

  設定ファイルには次のように書き込まれます。

  ```
  : Block as _match
  ```

- アクション部分に*複数*のアクションセットが含まれるルールを設定ファイルに書き込む場合、アクションが一覧表示されるとおりの順序で、同じ条件部分とアクション部分からのアクション1つがあるルールのシーケンスとして書き込まれます。

  たとえば、次のルールはアクション部分に*2つのアクション*を含みます。

  ```
  user in ('user1', 'user2') :SET http_templates_dir = "/etc/mytemplates",
  Block as _match
  ```

  設定ファイルには*2つのルールのシーケンス*として書き込まれます。

  ```
  user in ('user1', 'user2') :SET http_templates_dir = "/etc/mytemplates"
  user in ('user1', 'user2') : Block as _match
  ```

- ロギングまたはルールは、条件部分で条件の離接(論理和)を許可しないため、論理和を実装するには、各ルールに唯一の離接条件がある条件で一連のルールを記録します。たとえば、以下の2つのルールは、「悪意のあるオブジェクト*KnownVirus*またはカテゴリー*Terrorism*からのURLが検出された場合にブロックする」ルールと同等です。

  ```
  threat_category in (KnownVirus) : Block as _match
  url_category in (Terrorism) : Block as _match
  ```

  次のレコードは同等です。$(a \rightarrow x,\ b \rightarrow x)$、$((a \rightarrow x) \wedge (b \rightarrow x))$、$((a \vee b) \rightarrow x)$。

設定パラメータについては、管理ツールDr.Web Ctl(モジュール**drweb-ctl**)のコマンド`cfshow`と`cfset`を使用して、**`RuleSet`**などのパラメータの値(ルールなど)を表示、変更できます。コマンドライン管理ツールDr.Web Ctl(**drweb-ctl**モジュール)の`cfshow`と`cfset`のコマンド構文の詳細については、Dr.Web Ctlのセクションを参照してください。

付録E. SSL証明書を生成する

安全なSSL/TLSデータチャネルとHTTPS、LDAPS、SMTPSなどのアプリケーションプロトコルを使用するDr.Web for UNIX Internet Gatewaysコンポーネントの場合は、プライベートSSLキーと対応する証明書を提供する必要があります。一部のコンポーネントのキーと証明書は自動的に生成されます。それ以外の場合はDr.Web for UNIX Internet Gatewaysユーザーが作成する必要があります。すべてのコンポーネントがPEM形式の証明書を使用します。

認証局（CA）の検証証明書や署名付き証明書など、SSL / TLSを介した接続に使用されるプライベートキーと証明書を生成するには、コマンドラインユーティリティ**openssl**（**OpenSSL**暗号化パッケージに含まれる）を使用できます。

プライベートキーとそれに対応するSSL証明書を、CA検証証明書によって署名されたSSL証明書と共に生成するために必要な一連のアクションを検討します。

プライベートSSLキーと証明書を生成する

生成手順は2つのステップで構成されています。

1. プライベートキー（RSAアルゴリズム、キーの長さは2048ビット）を生成します。

   ```
   $ openssl genrsa -out keyfile.key 2048
   ```

   キーをパスワードで保護する場合は、`-des3`オプションを使用します。生成されたキーは、現在のディレクトリの`keyfile.key`ファイルにあります。キーを表示するには、次のコマンドを使用します。

   ```
   $ openssl rsa -noout -text -in keyfile.key
   ```

2. 既存のプライベートキーに基づいて指定された期間の証明書を生成します（この場合は365日）

   ```
   $ openssl req -new -x509 -days 365 -key keyfile.key -out certificate.crt
   ```

   このコマンドは、認証オブジェクトを識別するためのデータ（名前、組織など）を要求します。生成された証明書は、`certificate.crt`ファイルに置かれます。

   生成された証明書の内容をスキャンするには、次のコマンドを使用します。

   ```
   $ openssl x509 -noout -text -in certificate.crt
   ```

証明書を信頼済みCA証明書として登録する

信頼済みCA証明書のシステムリストに証明書を登録する場合（前の手順で証明書が生成される場合など）は、次の手順を実行します。

1. 証明書ファイルをシステムの信頼済み証明書ディレクトリ（**Debian/Ubuntu**の`/etc/ssl/certs/`）に移動またはコピーします。
2. 信頼済み証明書ディレクトリに、証明書へのシンボリックリンクを作成します。リンクの名前は証明書のハッシュ値です。
3. 証明書を含むシステムのディレクトリの内容にインデックスを付け直します。

以下の例では、これら3つすべてのアクションを実行します。ここでは、現在の証明書ディレクトリが信頼済み証明書ディレクトリ/etc/ssl/certs/であり、信頼済み証明書として登録されている証明書が/home/user/ca.crtファイルにあると想定しています。

```
# cp /home/user/ca.crt ./
# ln -s ca.crt `openssl x509 -hash -noout -in ca.crt`.0
# c_rehash /etc/ssl/certs/
```

署名付き証明書を作成する

署名付き証明書を作成するには、次の手順に従います。

1. 既存のプライベートキーに基づいて証明書に署名するためのリクエスト(*Certificate Signing Request (CSR)*)を生成します。キーが存在しない場合は、生成します。署名リクエストは次のコマンドで作成されます。

   ```
   $ openssl req -new -key keyfile.key -out request.csr
   ```

   このコマンドは、証明書を作成するコマンドと同様に、認証済みオブジェクトを識別するためのデータを要求します。このkeyfile.keyは、プライベートキーの既存のファイルです。受信したリクエストはrequest.csrファイルに保存されます。

   リクエストの作成結果を確認するには、次のコマンドを使用します。

   ```
   $ openssl req -noout -text -in request.csr
   ```

2. 次のコマンドを使用して、リクエストと既存のCA証明書に基づいて署名付き証明書を作成します。

   ```
   $ openssl x509 -req -days 365 -CA ca.crt -CAkey ca.key -set_serial 01 -in request.csr -out sigcert.crt
   ```

   署名付き証明書を作成するには、ルート証明書ca.crtとそのプライベートキーca.key(ca.crtとca.keyの代わりにcertificate.crt証明書とkeyfile.keyキーを使用することもできます。その場合、取得した証明書は自己署名されます)、署名リクエストのrequest.csrの3つのファイルが必要です。作成された署名付き証明書はsigcert.crtファイルに保存されます。

   結果を確認するには、次のコマンドを使用します。

   ```
   $ openssl x509 -noout -text -in sigcert.crt
   ```

作成する必要がある一意の証明書の数と同じだけこの手順を繰り返します。たとえば、スキャンクラスタ内の分散ファイルスキャンDr.Web Network Checkerのすべてのエージェントには、独自のキーと証明書が必要です。

署名付き証明書を変更する

一部のブラウザまたはメールクライアントでは、認証に使用される署名済み証明書をPKCS12形式に変更する必要があります。次のコマンドを使用して証明書を変更できます。

```
# openssl pkcs12 -export -in sigcert.crt -out sigcert.pfx -inkey keyfile.key
```

`Sigcert.crt`は署名済み証明書の既存ファイルです。`keyfile.key`は対応するプライベートキーのファイルです。変更した証明書は`sigcert.pfx`に保存されます。

付録F. 既知のエラー

このセクションには次が含まれます。

- エラーを特定するための推奨事項。
- エラーコードで検出されたエラーの説明。
- 症状によって検出された、コードのないエラーの説明。
- エラーコード、エラーメッセージ、それらの内部指定をリンクするカタログ。

本セクション内に記載されていないエラーが発生した場合は、テクニカルサポートに連絡することをお勧めします。その際、エラーコードと、問題を再現するための手順をお伝えください。

エラーを特定するための推奨事項

- エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって`/var/log/syslog`ファイルまたは`/var/log/messages`ファイルにあります)。また、コマンド **drweb-ctl** `log`を使用することもできます。
- エラーを特定するために、個別のファイルにログを記録するよう設定し、ログへの広範な情報の出力を有効にすることが推奨されます。そのために、以下のコマンドを実行してください。

```
# drweb-ctl cfset Root.Log <path to log file>
# drweb-ctl cfset Root.DefaultLogLevel DEBUG
```

- デフォルトのロギング方法とログの詳細レベルに戻すには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.Log -r
# drweb-ctl cfset Root.DefaultLogLevel -r
```

コードによるエラー

テキスト形式のエラーメッセージや数値のエラーコードを受け取る代わりに、`EC_XXX`のような内部エラーコード(たとえば、`EC_APP_TERMINATED`)を受信した場合は、エラーの内部カタログの表を使用して数値のエラーコードと、このセクションに記載されているエラーの説明を見つけることができます。

エラーメッセージ: *モニターチャネルに関するエラー*

エラーコード: `x1`

説明: コンポーネントの1つがDr.Web ConfigD設定デーモンと接続できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 以下のコマンドを実行することで設定デーモンを再起動させてください。

```
# service drweb-configd restart
```

2. **PAM** の認証メカニズムがインストール、設定されていて、正常に動作していることを確認します。そうでない場合は、インストール・設定します（詳細についてはお使いのOSディストリビューション向けの管理者ガイドとマニュアルを参照してください）。
3. **PAM** が正常に設定されていて、設定デーモンを再起動しても問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに復元してください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.iniファイルのコンテンツを削除します（設定ファイルのバックアップを作成することが推奨されます）。

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。
4. 設定デーモンを起動することができない場合は、drweb-configdパッケージを再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *操作はすでに実行中です。*

エラーコード: x2

説明: ユーザーによって要求された操作はすでに実行中です。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 操作が完了するまでお待ちください。必要に応じ、しばらく時間をおいて再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *操作は保留中です。*

エラーコード: x3

説明: ユーザーによって要求された操作は保留の状態です（ネットワーク接続を確立中、またはコンポーネントの1つがローディング中や初期化中で時間を要するなどの理由）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 操作が開始されるまでお待ちください。必要に応じ、しばらく時間をおいて再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ユーザーによって中断されました。*

エラーコード: x4

説明: アクションはユーザーによって終了されました(時間がかかるなどの理由)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. しばらく時間をおいて再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *操作がキャンセルされました。*

エラーコード: x5

説明: アクションがキャンセルされました(時間がかかるなどの理由によって)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 再度アクションを実行します。

引き続きエラーが発生する場合は、テクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *IPC接続が切断されました。*

エラーコード: x6

説明: Dr.Web for UNIX Internet Gatewaysコンポーネントの1つのプロセス間通信(IPC)が切断されました(多くの場合、ユーザーのコマンドによって、またはアイドル状態であるためにコンポーネントがシャットダウンしたことによって)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 操作が完了していない場合は、再度開始してください。そうでない場合、シャットダウンはエラーではありません。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なIPCメッセージサイズです。*

エラーコード: x7

説明: コンポーネントのプロセス間通信（IPC）中に無効なサイズのメッセージを受信しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysをリロードします。

```
# service drweb-configd restart
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なIPCメッセージフォーマットです。*

エラーコード: x8

説明: コンポーネントのプロセス間通信（IPC）中に無効なフォーマットのメッセージを受信しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysをリロードします。

```
# service drweb-configd restart
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *準備が完了していません。*

エラーコード: x9

説明: 必要なコンポーネントまたはデバイスがまだ初期化されていないため、要求されたアクションを実行できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. しばらく時間をおいて再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *コンポーネントがインストールされていません。*

エラーコード: x10

説明: 機能の実行に必要なコンポーネントがインストールされていません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 必要なコンポーネントをインストールまたは再インストールしてください。コンポーネントの名前が分からない場合は、ログファイルを確認して特定してください。
2. 必要なコンポーネントをインストールまたは再インストールしても解決しない場合は、Dr.Web for UNIX Internet Gatewaysを再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *予期せぬIPCメッセージです。*

エラーコード: x11

説明: コンポーネントのプロセス間通信(IPC)中に予期せぬメッセージを受信しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysをリロードします。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *IPCプロトコル違反です。*

エラーコード: x12

説明: コンポーネントのプロセス間通信(IPC)中にプロトコル違反が発生しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysをリロードします。

```
# service drweb-configd restart
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *サブシステムの状態が未知です。*

エラーコード: x13

説明: このソフトウェアの一部であり、要求された操作を実行するために必要な特定のサブシステムについて、その現在の状態が未知であるということが明らかになりました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 操作を繰り返します。
2. 引き続きエラーが発生する場合は、以下のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させてください。

   ```
   # service drweb-configd restart
   ```

   その後、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *パスは絶対パスでなければなりません。*

エラーコード: x20

説明: ファイルまたはディレクトリへの絶対パスが必要です(ファイルシステムのルートディレクトリから始まる)。現在使用されているのは相対パスです。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 絶対パスになるよう、ファイルまたはディレクトリへのパスを変更します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *十分なメモリがありません。*

エラーコード: x21

説明: 要求された操作(サイズの大きなファイルを開く、など)を完了するために必要な、十分なメモリがありません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあ

ります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. Dr.Web for UNIX Internet Gatewaysプロセスが使用可能なメモリのサイズを増やし（**ulimit**コマンドで上限を変更するなどして）、Dr.Web for UNIX Internet Gatewaysを再起動して操作を繰り返してください。

   場合によっては、システムサービス**systemd**は指定した上限の変更を無視できます。この場合、ファイル`/etc/systemd/system/drweb-configd .service.d/limits.conf`を編集し（ない場合は作成して）、変更後の制限値を指定します。たとえば、次のようになります。

   ```
   [Service]
   LimitDATA = 32767
   ```

   systemdの利用可能な上限のリストはドキュメント**man** `systemd.exec`で確認できます。

   次のコマンドを入力して、Dr.Web for UNIX Internet Gatewaysを再起動します。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*I/Oエラー*

エラーコード：`x22`

説明：入出力（I/O）エラーが発生しました（ドライブがまだ初期化されていない、またはファイルシステムのパーティションをもはや使用できない、など）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. 必要なI/Oデバイスまたはファイルシステムのパーティションが使用可能であるかどうかを確認します。必要に応じ、それをマウントして操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*指定されたファイルまたはディレクトリがありません。*

エラーコード：`x23`

説明：指定された、ファイルシステムのオブジェクト（ファイルまたはディレクトリ）がありません。削除された可能性があります。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. パスを確認します。必要に応じ、それを変更して操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *パーミッションが拒否されました。*

エラーコード: x24

説明: 指定された、ファイルシステムのオブジェクト（ファイルまたはディレクトリ）にアクセスする十分な権限がありません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. パスが正しいかどうか、また、コンポーネントが要求される権限を持っているかどうかを確認します。オブジェクトにアクセスする必要がある場合、アクセス権限を変更するか、コンポーネントの権限を昇格させます。操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ディレクトリではありません。*

エラーコード: x25

説明: ファイルシステムの、指定されたオブジェクトがディレクトリではありません。ディレクトリへのパスを入力してください。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. パスを確認します。それを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *データファイルが破損しています。*

エラーコード: x26

説明: 要求されたデータが破損しています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 操作を繰り返します。
2. 引き続きエラーが発生する場合は、以下のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させてください。

   ```
   # service drweb-configd restart
   ```

   その後、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ファイルはすでに存在しています。*

エラーコード: x27

説明: ファイルの作成を試みる際に、同じ名前を持つ別のファイルが検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. パスを確認します。それを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *読み取り専用ファイルシステム*

エラーコード: x28

説明: ファイルシステムのオブジェクト(ディレクトリ、ファイル、ソケット)を作成または変更しようと試みた際に、ファイルシステムが読み取り専用であることが明らかになりました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. パスを確認します。ファイルシステムの書き込み可能なパーティションを指すようにパスを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ネットワークエラー*

エラーコード: x29

説明: ネットワークエラーが発生しました(リモートホストが予期せず応答を停止したか、必要な接続に失敗した可能性があります)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ネットワークが使用可能であること、ネットワーク設定が正しいことを確認します。必要に応じ、ネットワーク設定を変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ドライブではありません。*

エラーコード: x30

説明: アクセスした入出力(I/O)がドライブではありません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ドライブ名を確認します。ドライブを指すようにパスを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *予期せぬEOFがあります。*

エラーコード: x31

説明: データの読み込み中に、予期せずファイルの末尾に達しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ファイル名を確認します。必要に応じ、正しいファイルを指すようにパスを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ファイルが変更されています。*

エラーコード: x32

説明: ファイルのスキャン中に、ファイルが変更されていることが明らかになりました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 再スキャンしてください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *通常ファイルではありません。*

エラーコード: x33

説明: ファイルシステムのオブジェクトへのアクセス中に、それが通常ファイルではないということが明らかになりました(つまり、ディレクトリ、ソケット、またはファイルシステムのその他のオブジェクトである)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. ファイル名を確認します。必要に応じ、通常ファイルを指すようにパスを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*名前はすでに使用されています*

エラーコード：`x34`

説明：ファイルシステムのオブジェクト（ディレクトリ、ファイル、ソケット）の作成を試みた際に、同じ名前を持つ別のオブジェクトが検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. パスを確認します。それを変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*ホストがオフラインです。*

エラーコード：`x35`

説明：ネットワーク経由でリモートホストを利用できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. 必要なホストが使用可能であるかどうかを確認します。必要に応じ、ホストアドレスを変更して操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*リソースの上限に達しています。*

エラーコード：`x36`

説明：特定のリソースの使用について設定された上限に達しています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 必要なリソースの使用可能状況を確認します。必要に応じ、このリソースの使用に関する上限を引き上げて、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *異なるマウントポイントです。*

エラーコード: x37

説明: 異なるマウントポイントに属するファイルシステムディレクトリ間の移動を必要とするファイル復元の試みです。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ファイルを復元する別のパスを選択し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *アンパックエラー*

エラーコード: x38

説明: アーカイブのアンパックに失敗しました(パスワード保護されているか、破損している可能性があります)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ファイルが破損していないことを確認します。アーカイブがパスワード保護されている場合、正しいパスワードを入力することで保護を解除し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ウイルスデータベースが破損しています。*

エラーコード: x40

説明: ウイルスデータベースが破損しています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ウイルスデータベースディレクトリへのパスを確認してください。必要に応じてパスを変更します(設定ファイ

ルの[Root]セクションにあるVirusBaseDir**VirusBaseDir**パラメータ)。

パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。

または、コマンドライン管理ツールのコマンドを使用することもできます。

現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow Root.VirusBaseDir
```

新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *サポートされていないバージョンのウイルスデータベースです。*

エラーコード: x41

説明: 現在のウイルスデータベースはプログラムの古いバージョン向けのものです。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl**logを使用することもできます。

エラーの解決:

1. ウイルスデータベースディレクトリへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[Root]セクションにあるVirusBaseDir**VirusBaseDir**パラメータ)。

   パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。

   または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow Root.VirusBaseDir
```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ウイルスデータベースが空です。*

エラーコード: x42

説明: ウイルスデータベースが空です。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ウイルスデータベースディレクトリへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[Root]セクションにあるVirusBaseDir**VirusBaseDir**パラメータ)。
   - パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow Root.VirusBaseDir
```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir <new path>
```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.VirusBaseDir -r
```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *オブジェクトを修復できません。*

エラーコード: x43

説明: 脅威の駆除中に、修復不可能なオブジェクトに対して「修復」アクションを適用する試みが行われました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. オブジェクトに対して適用可能なアクションを選択し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *サポートされていないウイルスデータベースの組み合わせです。*

エラーコード: x44

説明: 現在のウイルスデータベースの組み合わせはサポートされていません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. ウイルスデータベースディレクトリへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[Root]セクションにあるVirusBaseDir**VirusBaseDir**パラメータ)。
   - パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Root.VirusBaseDir
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Root.VirusBaseDir <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Root.VirusBaseDir -r
   ```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

   ```
   $ drweb-ctl update
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *スキャンの上限に達しています。*

エラーコード: x45

説明: オブジェクトのスキャン中に、指定された上限に達しました(アンパックされたファイルのサイズ上限、ネスティングレベルの上限など)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 以下のいずれかの方法で、スキャンにおける上限を変更します(コンポーネント設定内で)。
   - Webインターフェースのコンポーネント設定のあるページ(インストールされている場合)。
   - **drweb-ctl** cfshowおよび**drweb-ctl** cfsetコマンドを使用します。
2. 設定の変更後、試みた操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *認証に失敗しました。*

エラーコード: x47

説明: 認証に、無効なユーザー認証情報が使用されています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 必要な権限を持ったユーザーの有効な認証情報を入力してください。再度、認証を実行してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *認証に失敗しました。*

エラーコード: x48

説明: 認証に使用された認証情報を持つユーザーが十分な権限を有していません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 必要な権限を持ったユーザーの有効な認証情報を入力してください。再度、認証を実行してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なアクセストークンです。*

エラーコード: x49

説明: Dr.Web for UNIX Internet Gatewaysコンポーネントの1つが、昇格された権限を必要とする操作へのアクセスを試みる際に無効な認証トークンを提示しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 必要な権限を持ったユーザーの有効な認証情報を入力してください。再度、認証を実行してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な引数です。*

エラーコード: x60

説明: コマンドの実行を試みる際に、無効な引数が使用されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 有効な引数を使用して、再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な操作です。*

エラーコード: x61

説明: 無効なコマンドを実行しようとする試みが検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 有効なコマンドを使用して、再度アクションを実行します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *スーパーユーザー権限が必要です。*

エラーコード: x62

説明: このアクションを実行することができるのは、スーパーユーザー権限を持ったユーザーのみです。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 権限をルート権限に昇格させ、再度アクションを実行します。権限を昇格させるには、**su** および **sudo** コマンドを使用します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *集中管理モードでは許可されていません。*

エラーコード: `x63`

説明: 要求されたアクションは、Dr.Web for UNIX Internet Gatewaysがスタンドアロンモードで動作している場合のみ実行できます。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. Dr.Web for UNIX Internet Gatewaysの動作モードをスタンドアロンモードに変更し、操作を繰り返します。
2. Dr.Web for UNIX Internet Gatewaysをスタンドアロンモードに切り替えるには
   - Webインターフェースがインストールされている場合は、Webインターフェースの**集中管理**の**集中管理モードを有効にする**チェックボックスをオフにします。
   - または、コマンド を実行します。

```
# drweb-ctl esdisconnect
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *サポートされていないOSです。*

エラーコード: `x64`

説明: Dr.Web for UNIX Internet Gatewaysは、ホスト上にインストールされているOSをサポートしていません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. システム要件のリスト内に記載されているOSをインストールします。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *実装されていない機能です。*

エラーコード: x65

説明: コンポーネントの1つの、必要な機能がプログラムの現在のバージョンには備わっていません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. Dr.Web for UNIX Internet Gatewaysの設定をデフォルトに復元します。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.iniファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます)。

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に、次のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させます。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *未知のオプションです。*

エラーコード: x66

説明: 設定ファイルに、未知のパラメータまたはDr.Web for UNIX Internet Gatewaysの現在のバージョンでサポートされていないパラメータが含まれています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. いずれかのテキストエディターで*<etc_dir>*/drweb.iniファイルを開き、無効なパラメータが含まれる行を削除します。ファイルを保存し、次のコマンドを実行することでDr.Web ConfigD設定デーモンを再起動させます。

   ```
   # service drweb-configd restart
   ```

2. 問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してしてください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます):

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *未知のセクションです。*

エラーコード: x67

説明: 設定ファイルに、未知のセクションまたはDr.Web for UNIX Internet Gatewaysの現在のバージョンでサポートされていないセクションが含まれています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. いずれかのテキストエディターで *<etc_dir>*/drweb.iniファイルを開き、未知の(サポートされていない)セクションを削除します。ファイルを保存し、次のコマンドを実行することでDr.Web ConfigD設定デーモンを再起動させます。

   ```
   # service drweb-configd restart
   ```

2. 問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してしてください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます):

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なオプション値です。*

エラーコード: x68

説明: 設定ファイル内のパラメータの1つに、無効な値が含まれています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 以下のいずれかの方法で、有効なパラメータ値を設定します。
   - Webインターフェースのコンポーネント設定のあるページ(インストールされている場合)。
   - **drweb-ctl** cfshowおよび**drweb-ctl** cfsetコマンドを使用します。

   当該パラメータの有効な値が分からない場合は、そのパラメータを使用するコンポーネントのヘルプファイルを参照してください。パラメータ値をデフォルト値に戻すこともできます。
2. 設定ファイル *<etc_dir>*/drweb.iniを直接編集することも可能です。その場合は、いずれかのテキストエディターで設定ファイルを開き、無効なパラメータ値を含む行を見つけ、有効な値を設定してください。その後、ファイルを保存し、以下のコマンドを実行することでDr.Web ConfigD設定デーモンを再起動させます。

```
# service drweb-configd restart
```

3. この手順で問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します（設定ファイルのバックアップを作成することが推奨されます）：

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

   設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な状態です。*

エラーコード: x69

説明: Dr.Web for UNIX Internet Gatewaysまたはコンポーネントの1つが無効な状態であるため、必要な操作を完了できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. しばらく時間をおいて再度アクションを実行します。
2. 引き続きエラーが発生する場合は、以下のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させてください。

```
# service drweb-configd restart
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *使用可能な値は1つのみです。*

エラーコード: x70

説明: 設定ファイル内のパラメータの1つに、値のリストが含まれています。含むことができるのは1つの値のみです。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 以下のいずれかの方法で、有効なパラメータ値を設定します。
   - Webインターフェースのコンポーネント設定のあるページ（インストールされている場合）。
   - **drweb-ctl** cfshowおよび**drweb-ctl** cfsetコマンドを使用します。

   当該パラメータの有効な値が分からない場合は、そのパラメータを使用するコンポーネントのヘルプファイ

ルを参照してください。パラメータ値をデフォルト値に戻すこともできます。

2. 設定ファイル *<etc_dir>*/drweb.iniを直接編集することも可能です。その場合は、いずれかのテキストエディターで設定ファイルを開き、無効なパラメータ値を含む行を見つけ、有効な値を設定してください。その後、ファイルを保存し、以下のコマンドを実行することでDr.Web ConfigD設定デーモンを再起動させます。

```
# service drweb-configd restart
```

3. この手順で問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します（設定ファイルのバックアップを作成することが推奨されます）：

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

   設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なタグ値です。*

エラーコード: x71

説明: ユニークなタグ識別子を含んだ名前を持つ設定ファイル内のセクションの1つに、無効なタグ識別子があります。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. Webインターフェース内にセクションを作成しようとした際、またはコマンドの使用中にエラーが発生した場合

```
# drweb-ctl cfset <section>.<parameter> <new value>
```

   タグに有効な値を設定し、セクションを再度保存します。

2. セクションが設定ファイル *<etc_dir>*/drweb.iniに直接保存されている場合は、ファイルを編集します。これを行うには、いずれかのテキストエディターで設定ファイルを開き、無効なタグ値を含むセクション名を見つけて、タグに有効な値を設定します。ファイルを保存し、次のコマンドを実行することでDr.Web ConfigD設定デーモンを再起動させます。

```
# service drweb-configd restart
```

3. この手順で問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します（設定ファイルのバックアップを作成することが推奨されます）：

```
# cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
# echo "" > /etc/opt/drweb.com/drweb.ini
```

設定ファイルのコンテンツを削除した後に設定デーモンを再起動させます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *レコードが見つかりません。*

エラーコード: x80

説明: 脅威のレコードにアクセスしようとした際に、レコードがないことが明らかになりました(脅威を処理したのが別のDr.Web for UNIX Internet Gatewaysコンポーネントであった可能性があります)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. しばらくしてから脅威のリストを更新してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *レコードは現在処理中です。*

エラーコード: x81

説明: 脅威のレコードにアクセスしようとした際に、レコードが別のコンポーネントによって処理中であることが明らかになりました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. しばらくしてから脅威のリストを更新してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ファイルはすでに隔離済みです。*

エラーコード: x82

説明: 検出された脅威を含むファイルを隔離に移動しようとした際に、ファイルがすでに隔離されていることが明らかになりました(脅威が別のコンポーネントによって処理された可能性があります)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. しばらくしてから脅威のリストを更新してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *更新前にバックアップを行うことができません。*

エラーコード: x89

説明: 更新サーバーから更新をダウンロードする前に対象となるファイルのバックアップコピーを作成しようとする試みが失敗しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. 更新されたファイルのバックアップコピーを保存するディレクトリへのパスを確認します。必要に応じてパスを変更します(設定ファイルの `[Update]`セクションにある**BackupDir**パラメータ)。
   - パスを表示および変更するには、Webインターフェースの **Updater** ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Update.BackupDir
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Update.BackupDir <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Update.BackupDir -r
   ```
2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

   ```
   $ drweb-ctl update
   ```
3. 引き続きエラーが発生する場合は、Dr.Web Updaterコンポーネントを実行しているアカウントのユーザーが、**BackupDir**で指定されたディレクトリへの書き込み権限を持っているかどうかを確認してください。このユーザーの名前は、**RunAsUser**パラメータで指定されます。必要に応じて、**RunAsUser**パラメータで指定されているユーザーを変更するか、ディレクトリのプロパティで足りない権限を付与します。
4. それでもエラーが続く場合は、`drweb-update`パッケージを再インストールします。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法についてはDr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なDRLファイルです。*

エラーコード: x90

説明: 更新サーバーのリストが含まれているファイルの1つで、整合性違反が検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. サーバーのリストを含むファイルへのパスを確認し、必要に応じてパスを変更します（設定ファイルの `[Update]`セクションにある**`*DrlDir`**を持つパラメータ）。
   - パスを表示および変更するには、Webインターフェースの **Updater** ページに移動します（インストールされている場合）。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を表示するには、コマンドを使用します（*<*DrlDirPath>* は、指定されたパラメータ名に置き換える必要があります。パラメータ名が不明な場合は、セクション内のパラメータ値を参照します。角括弧内のコマンド部分は省略します。

   ```
   $ drweb-ctl cfshow Update[.<*DrlDir>]
   ```

   新しいパラメータ値を設定するには、コマンドを実行します（*<*DrlDir>* は、指定されたパラメータ名に置き換える必要があります。

   ```
   # drweb-ctl cfset Update.<*DrlDir> <new path>
   ```

   パラメータ値をデフォルトに戻すには、コマンドを実行します（*<*DrlDir>* は、指定されたパラメータ名に置き換える必要があります。

   ```
   # drweb-ctl cfset Update.<*DrlDir> -r
   ```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

   ```
   $ drweb-ctl update
   ```

3. それでもエラーが続く場合は、`drweb-update`パッケージを再インストールします。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*無効なLSTファイルです。*

エラーコード：`x91`

説明：更新されたウイルスデータベースのリストが含まれているファイルで、整合性違反が検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. しばらく時間をおいて再度ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

   ```
   $ drweb-ctl update
   ```

2. それでもエラーが続く場合は、`drweb-update`パッケージを再インストールします。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な圧縮ファイルです。*

エラーコード: `x92`

説明: 更新が含まれているダウンロードされたファイルで、整合性違反が検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. しばらく時間をおいて再度ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

   ```
   $ drweb-ctl update
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *プロキシ認証エラーです。*

エラーコード: `x93`

説明: プログラムは、設定内で指定されたプロキシサーバーを使用して更新サーバーに接続できませんでした。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. プロキシサーバーへの接続に使用されているパラメータを確認します(設定ファイルの `[Update]`セクションの**Proxy**パラメータで設定されています)。

- 接続パラメータを表示および設定するには、Webインターフェースの **Updater** ページに移動します（インストールされている場合）。
- または、コマンドライン管理ツールのコマンドを使用することもできます。

現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow Update.Proxy
```

新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Update.Proxy <new parameters>
```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Update.Proxy -r
```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

   ```
   $ drweb-ctl update
   ```

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *使用可能な更新サーバーがありません。*

エラーコード: x94

説明: プログラムは、いずれの更新サーバーにも接続できませんでした。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. ネットワークが使用可能であるかどうかを確認します。必要に応じ、ネットワーク設定を変更します。
2. プロキシサーバーのみを使用してネットワークにアクセスできる場合は、プロキシサーバーに接続するためのパラメータを設定します（設定ファイルの `[Update]`セクションの**Proxy**パラメータで設定できます）。
   - 接続パラメータを表示および設定するには、Webインターフェースの **Updater** ページに移動します（インストールされている場合）。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Update.Proxy
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Update.Proxy <new parameters>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Update.Proxy -r
```

3. ネットワーク接続パラメータ(プロキシサーバーのパラメータを含む)が正しくてもエラーが発生する場合は、利用可能な更新サーバーのリストを使用していることを確認してください。使用されている更新サーバーのリストは、設定ファイルの [Update] セクションのパラメータ***DrlDir**で表示されます。
CustomDrlDir**パラメータが既存の正しいサーバーリストのファイルを示す場合、標準的な更新ゾーンのサーバーではなく、リスト内で指定されたサーバーが使用されます(対応するDrlDir**パラメータで指定されている値は無視されます)。
   - 接続パラメータを表示および設定するには、Webインターフェースの **Updater** ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を表示するには、コマンドを使用します(*<*DrlDirPath>* は、指定されたパラメータ名に置き換える必要があります。パラメータ名が不明な場合は、セクション内のパラメータ値を参照します。角括弧内のコマンド部分は省略します。

   ```
   $ drweb-ctl cfshow Update[.<*DrlDir>]
   ```

   新しいパラメータ値を設定するには、コマンドを実行します(*<*DrlDir>* は、指定されたパラメータ名に置き換える必要があります。

   ```
   # drweb-ctl cfset Update.<*DrlDir> <new path>
   ```

   パラメータ値をデフォルトに戻すには、コマンドを実行します(*<*DrlDir>* は、指定されたパラメータ名に置き換える必要があります。

   ```
   # drweb-ctl cfset Update.<*DrlDir> -r
   ```

4. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

   ```
   $ drweb-ctl update
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *キーファイルのフォーマットが無効です。*

エラーコード: x95

説明: キーファイルのフォーマットがサポートされていません

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. キーファイルを持っているかどうか、また、キーファイルへのパスを確認します。キーファイルへのパスは、設定ファイルの [Root]セクションの**KeyPath**パラメータで指定できます。
   - キーファイルへのパスを表示および設定するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。

- または、コマンドライン管理ツールのコマンドを使用することもできます。

  現在のパラメータ値を確認するには、以下のコマンドを実行します。

  ```
  $ drweb-ctl cfshow Root.KeyPath
  ```

  新しいパラメータ値を設定するには、以下のコマンドを実行します。

  ```
  # drweb-ctl cfset Root.KeyPath <path to file>
  ```

  パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

  ```
  # drweb-ctl cfset Root.KeyPath -r
  ```

2. キーファイルをお持ちでない場合、または使用しているキーファイルが破損している場合は、キーファイルを購入してインストールしてください。キーファイル、購入、インストールに関する詳細については、ライセンスを参照してください。
3. キーファイルをインストールするには、Webインターフェースの**メイン**ページの下部にあるライセンス有効化フォームを使用できます(インストールされている場合)。
4. また、https://support.drweb.com/get+cabinet+link/のユーザーのWebページ**My Dr.Web**で、現在のライセンスオプションを確認できます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ライセンスは有効期限が切れています。*

エラーコード: x96

説明: 使用しているライセンスは有効期限が切れています。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 新しいライセンスを購入して受け取るキーファイルをインストールします。ライセンスの購入方法とキーファイルのインストールの詳細については、管理者マニュアルのライセンスを参照してください。
2. 購入したキーファイルをインストールするには、Webインターフェースの**メイン**ページの下部にあるライセンス有効化フォームを使用できます(インストールされている場合)。
3. また、https://support.drweb.com/get+cabinet+link/のユーザーのWebページ**My Dr.Web**で、現在のライセンスオプションを確認できます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ネットワークオペレーションのタイムアウト。*

エラーコード: x97

説明: ネットワークオペレーションがタイムアウトしました(リモートホストが予期せず応答を停止したか、必要な接続に失敗した可能性があります)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあ

ります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. ネットワークが使用可能であること、ネットワーク設定が正しいことを確認します。必要に応じ、ネットワーク設定を変更し、操作を繰り返します。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*無効なチェックサムです。*

エラーコード：`x98`

説明：更新が含まれているダウンロードされたファイルのチェックサムが検出されました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. しばらく時間をおいて再度ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンド を実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*無効なデモキーファイルです。*

エラーコード：`x99`

説明：使用されているデモキーファイルが無効です（別のコンピューターから受け取ったものであるなど）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. 該当するコンピューターの新しい試用期間のリクエストを送信するか、新しいライセンスを購入して、受け取るキーファイルをインストールしてください。ライセンスの購入方法とキーファイルのインストールの詳細については、管理者マニュアルのライセンスを参照してください。
2. 購入したキーファイルをインストールするには、Webインターフェースの**メイン**ページの下部にあるライセンス有効化フォームを使用できます（インストールされている場合）。
3. また、https://support.drweb.com/get+cabinet+link/のユーザーのWebページ**My Dr.Web**で、現在のライセンスオプションを確認できます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ライセンスキーファイルがブロックされています。*

エラーコード: x100

説明: 使用されているライセンスはブロックされています（Dr.Web for UNIX Internet Gatewaysの使用に関するライセンス使用許諾契約の条件に違反している可能性があります）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 新しいライセンスを購入して受け取るキーファイルをインストールします。ライセンスの購入方法とキーファイルのインストールの詳細については、管理者マニュアルのライセンスを参照してください。
2. 受領したキーファイルをインストールするには、Webインターフェースの**メイン**ページの下部にあるライセンス有効化フォームを使用できます（インストールされている場合）。
3. また、https://support.drweb.com/get+cabinet+link/のユーザーのWebページ**My Dr.Web**で、現在のライセンスオプションを確認できます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効なライセンスです。*

エラーコード: x101

説明: 使用されているライセンスが別の製品のものであるか、Dr.Web for UNIX Internet Gatewaysコンポーネントの動作がライセンスで許可されていません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 新しいライセンスを購入して受け取るキーファイルをインストールします。ライセンスの購入方法とキーファイルのインストールの詳細については、管理者マニュアルのライセンスを参照してください。
2. 受領したキーファイルをインストールするには、Webインターフェースの**メイン**ページの下部にあるライセンス有効化フォームを使用できます（インストールされている場合）。
3. また、https://support.drweb.com/get+cabinet+link/のユーザーのWebページ**My Dr.Web**で、現在のライセンスオプションを確認できます。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な設定です。*

エラーコード: x102

説明: Dr.Web for UNIX Internet Gatewaysコンポーネントの1つが、誤った設定のために動作できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります）。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. エラーが発生しているコンポーネントの名前が分からない場合は、ログファイルを確認して特定してください。
2. このエラーがDr.Web Firewall for Linuxによって発生している場合、他のファイアウォールと競合している可能性があります。たとえば、Dr.Web Firewall for Linuxが**Fedora**、**CentOS**、**Red Hat Enterprise Linux**の**FirewallD**と競合することが知られています(起動するたびに、**FirewallD**はDr.Web Firewall for Linuxにより追加されたトラフィックルーティングルールを破棄します)。このエラーを解決するには、以下のコマンドを実行してDr.Web for UNIX Internet Gatewaysを再起動します

   ```
   # service drweb-configd restart
   ```

   または

   ```
   # drweb-ctl reload
   ```

   FirewallDの動作を許可すると、OSの再起動も含め、**FirewallD**の再起動ごとにDr.Web Firewall for Linuxのエラーが繰り返し発生する可能性があります。このエラーを解決するには、**FirewallD**を無効にします(お使いのOSのマニュアルに含まれている**FirewallD**のマニュアルを参照してください)。

3. エラーが別のコンポーネントによって発生している場合、以下のいずれかの方法によって、そのコンポーネントの設定をデフォルトに復元してください。
   - Webインターフェースのコンポーネント設定のあるページ(インストールされている場合)。
   - **drweb-ctl** `cfshow`および**drweb-ctl** `cfset`コマンドを使用します。
   - 手動で設定ファイルを編集してください(コンポーネントセクションからすべてのパラメータを削除してください)。
4. この手順で問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*`/drweb.ini` ファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます):

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に、次のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させます。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *無効な実行ファイルです。*

エラーコード: `x104`

説明: パスが誤っているか、または実行ファイルのコンテンツが破損していることが原因で、Dr.Web for UNIX Internet Gatewaysコンポーネントの1つを実行できません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. エラーが発生しているコンポーネントの名前が分からない場合は、ログファイルを確認して特定してください。
2. 以下のコマンドを実行することで（*<component section>*を、設定ファイルの該当するセクション名に変更します）、Dr.Web for UNIX Internet Gateways設定ファイル内でコンポーネントの実行可能パスを確認してください（コンポーネントセクションの**ExePath**パラメータ）。

   ```
   $ drweb-ctl cfshow <component section>.ExePath
   ```

3. 以下のコマンドを実行することで（*<component section>*を、設定ファイルの該当するセクション名に変更します）、パスをデフォルトに復元します。

   ```
   # drweb-ctl cfset <component section>.ExePath -r
   ```

4. この手順で問題が解決しない場合は、該当するコンポーネントのパッケージを再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ：*Virus-Finding Engineを使用することができません*

エラーコード：`x105`

説明：Dr.Web Virus-Finding Engineのファイルが見つからないか使用できません（脅威の検出に必要です）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. **drweb32.dll**スキャンエンジンファイルへのパスを確認します。必要に応じてパスを変更します（設定ファイルの[Root]セクションにある**CoreEnginePath**パラメータ）。
   - パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します（インストールされている場合）。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Root.CoreEnginePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Root.CoreEnginePath <new path>
   ```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.CoreEnginePath -r
```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

   ```
   $ drweb-ctl update
   ```

3. パスが正しく、ウイルスデータベースを更新した後もエラーが続く場合は、`drweb-bases` パッケージを再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ウイルスデータベースがありません。*

エラーコード: `x106`

説明: ウイルスデータベースが見つかりません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. ウイルスデータベースディレクトリへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[`Root`]セクションにあるVirusBaseDir**`VirusBaseDir`**パラメータ)。
   - パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Root.VirusBaseDir
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Root.VirusBaseDir <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Root.VirusBaseDir -r
   ```

2. ウイルスデータベースを更新します。
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *プロセスはシグナルによって中断されました。*

エラーコード: x107

説明: コンポーネントがシャットダウンしました(ユーザーコマンドによって、またはアイドル状態であるためである可能性があります)

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 操作が完了していない場合は、再度開始してください。そうでない場合、シャットダウンはエラーではありません。
2. コンポーネントが度々シャットダウンする場合は、以下のいずれかの方法によって、そのコンポーネントの設定をデフォルトに復元します。
   - Webインターフェースのコンポーネント設定のあるページ(インストールされている場合)。
   - **drweb-ctl** cfshowおよび**drweb-ctl** cfsetコマンドを使用します。
   - 手動で設定ファイルを編集してください(コンポーネントセクションからすべてのパラメータを削除してください)。
3. 問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してしてください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*/drweb.ini ファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます):

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に、次のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させます。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *予期せぬプロセスの中断です。*

エラーコード: x108

説明: 不具合によって予期せずコンポーネントがシャットダウンしました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. 中断された操作を繰り返します。
2. コンポーネントが度々異常にシャットダウンする場合は、以下のいずれかの方法によって、そのコンポーネントの設定をデフォルトに復元します。
   - Webインターフェースのコンポーネント設定のあるページ(インストールされている場合)。
   - **drweb-ctl** `cfshow`および**drweb-ctl** `cfset`コマンドを使用します。
   - 手動で設定ファイルを編集してください(コンポーネントセクションからすべてのパラメータを削除してください)。
3. 問題が解決しない場合は、Dr.Web for UNIX Internet Gateways設定をデフォルトに戻してしてください。

   そのために、たとえば以下のコマンドを実行するなどして、*<etc_dir>*`/drweb.ini` ファイルのコンテンツを削除します(設定ファイルのバックアップを作成することが推奨されます):

   ```
   # cp /etc/opt/drweb.com/drweb.ini /etc/opt/drweb.com/drweb.ini.save
   # echo "" > /etc/opt/drweb.com/drweb.ini
   ```

   設定ファイルのコンテンツを削除した後に、次のコマンドを実行することでDr.Web for UNIX Internet Gatewaysを再起動させます。

   ```
   # service drweb-configd restart
   ```
4. Dr.Web for UNIX Internet Gateways設定を復元した後もエラーが続く場合は、コンポーネントパッケージを再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法についてはDr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *互換性のないソフトウェアが検出されました。*

エラーコード: `x109`

説明: 互換性のないソフトウェアが検出されたため、Dr.Web for UNIX Internet Gatewaysコンポーネントは動作していません。このソフトウェアはコンポーネントの正しい動作を中断します。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. このエラーがSpIDer Gateによって発生している場合、オペレーティングシステムに互換性のないソフトウェアがある可能性があります。このソフトウェアは、**NetFilter**システムファイアウォールのルールを生成します。これにより、SpIDer Gateが正しく動作しなくなります。おそらく、**Shorewall**または**SuseFirewall2**がシステムにインストールされているでしょう(**SUSE Linux** OSの場合)。**NetFilter**システムファイアウォールを設定するアプリケーションは、指定されたルールシステムの整合性をチェックして書き換えることがあります。これがSpIDer Gateがこのようなアプリケーションと競合する主な理由です。

   SpIDer Gate操作に干渉しないように、互換性のないソフトウェアを再設定してください。それが不可能な場合は、オペレーティングシステムの起動時に読み込まれないようにソフトウェアを無効にします。以下の手順に従って、**SuseFirewall2** アプリケーション(**SUSE Linux** OSの場合)の設定を行ってください。

   1) **SuseFirewall2**の設定ファイル(デフォルトでは `/etc/sysconfig/SuSEfirewall2`ファイルです)

を開きます。

2) 以下のテキストブロックを見つけます。

```
# Type: yesno
#
# Install NOTRACK target for interface lo in the raw table. Doing so
# speeds up packet processing on the loopback interface. This breaks
# certain firewall setups that need to e.g. redirect outgoing
# packets via custom rules on the local machine.
#
# Defaults to "yes" if not set
#
FW_LO_NOTRACK=""
```

3) パラメータ値に `No` を指定します。

```
FW_LO_NOTRACK="no"
```

4) **SuseFirewall2**を再起動します。そのために、以下のコマンドを使用します。

```
# rcSuSEfirewall2 restart
```

SuseFirewall2の設定内に`FW_LO_NOTRACK`がない場合、競合を解決するためにはアプリケーションを無効にし、それがシステム起動時に読み込まれないようにする必要があるという点に注意してください(たとえば、**SUSE Linux Enterprise Server** 11にはこれが必要です)。

競合アプリケーションを再設定または無効にした後、SpIDer Gateを再起動してください。

2. エラーが別のコンポーネントによって発生している場合、互換性のないソフトウェアを無効にするか再設定し、Dr.Web for UNIX Internet Gatewaysの動作を妨げないようにしてください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *Webリソースカテゴリーのデータベースがありません。*

エラーコード: `x112`

説明: Webリソースカテゴリーのデータベースがありません。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. Webリソースカテゴリーディレクトリのデータベースへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの`[Root]`セクションにある**DwsDir**パラメータ)。
   - パスを表示および変更するには、Webインターフェースの**全般設定**ページに移動します(インストールされている場合)。
   - または、コマンドライン管理ツールのコマンドを使用することもできます。

現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow Root.DwsDir
```

新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.DwsDir <new path>
```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset Root.DwsDir -r
```

2. Webリソースカテゴリーのデータベースを更新:
   - Webインターフェースがインストールされている場合は、Webインターフェースの**メイン**ページで**更新**をクリックします。
   - または、コマンドを実行します。

```
$ drweb-ctl update
```

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *LookupDを使用することができません。*

エラーコード: x115

説明: Dr.Web LookupDコンポーネントが見つかりません(外部ソースからデータを選択する必要があります)

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. **drweb-lookupd**実行ファイルへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[LookupD]セクションにある**ExePath**パラメータ)。

   または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow LookupD.ExePath
```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset LookupD.ExePath <new path>
```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset LookupD.ExePath -r
```

2. 設定にDr.Web LookupDコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、drweb-lookupdパッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法につ

いてはDr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gateways をアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *SpIDer Gateを使用することができません。*

エラーコード: x117

説明: SpIDer Gateコンポーネントがありません(ネットワーク接続のスキャンに必要です)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. **drweb-gated**実行ファイルへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[GateD]セクションにある**ExePath**パラメータ)。

   または、コマンドライン管理ツールの コマンド を使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow GateD.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset GateD.ExePath <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset GateD.ExePath -r
   ```

2. 設定にSpIDer Gateコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、drweb-gatedパッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gateways をアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *Scanning Engineを使用することができません。*

エラーコード: x119

説明: Dr.Web Scanning Engineコンポーネントが見つからないか、起動に失敗しました(脅威の検出に必要です)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. **drweb-se**実行ファイルへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[ScanEngine]セクションにある**ExePath**パラメータ)。

   または、コマンドライン管理ツールの コマンド を使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow ScanEngine.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset ScanEngine.ExePath <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset ScanEngine.ExePath -r
   ```

2. 正しいパスを入力した後もエラーが継続する場合は
   - 次のコマンドを実行します。

     ```
     $ drweb-ctl rawscan /
     ```

     行「エラー: 有効なライセンスが指定されていません」が出力された場合、有効なキーファイルがありません。Dr.Web for UNIX Internet Gatewaysを登録してライセンスを取得します。ライセンスを取得したら、キーファイルが使用可能かどうかを確認し、必要に応じてインストールします。
   - お使いのOSが**SELinux**を使用している場合、**drweb-se**モジュールに対するセキュリティポリシーを設定します(管理者マニュアルのSELinux セキュリティポリシーを設定するを参照してください)。
3. 設定にDr.Web Scanning Engineコンポーネントの設定が含まれていない場合、またはこれまでの手順で問題が解決しない場合は、`drweb-se`パッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *File Checkerを使用することができません。*

エラーコード: `x120`

説明: Dr.Web File Checkerコンポーネントが見つからないか、起動に失敗しました(脅威の検出に必要です)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります)。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. **drweb-filecheck**実行ファイルへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[FileCheck]セクションにある**ExePath**パラメータ)。

   または、コマンドライン管理ツールの コマンド を使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

```
$ drweb-ctl cfshow FileCheck.ExePath
```

新しいパラメータ値を設定するには、以下のコマンドを実行します。

```
# drweb-ctl cfset FileCheck.ExePath <new path>
```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset FileCheck.ExePath -r
```

2. 正しいパスを入力した後もエラーが継続する場合は
   - お使いのOSが**SELinux**を使用している場合、**drweb-filecheck**モジュールに対するセキュリティポリシーを設定します（管理者マニュアルのSELinux セキュリティポリシーを設定するを参照してください）。
3. 設定にDr.Web File Checkerコンポーネントの設定が含まれていない場合、またはこれまでの手順で問題が解決しない場合は、`drweb-filecheck`パッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法についてはDr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *ES Agentを使用することができません。*

エラーコード: `x121`

説明: Dr.Web ES Agentコンポーネントがありません（集中管理サーバーへの接続に必要です）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決:

1. **drweb-esagent**実行ファイルへのパスを確認してください。必要に応じてパスを変更します（設定ファイルの`[ESAgent]`セクションにある**ExePath**パラメータ）。

   または、コマンドライン管理ツールの コマンド を使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow ESAgent.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset ESAgent.ExePath <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset ESAgent.ExePath -r
   ```

2. 設定にDr.Web ES Agentコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、`drweb-esagent`パッケージをインストールまたは再インストールしてください。

Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *Firewall for Linuxを使用することができません。*

エラーコード: x122

説明: Dr.Web Firewall for Linuxコンポーネントがありません(ネットワーク接続のスキャンに必要です)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンド**drweb-ctl** logを使用することもできます。

エラーの解決:

1. **drweb-firewall**実行ファイルへのパスを確認してください。必要に応じてパスを変更します(設定ファイルの[LinuxFirewall]セクションにある**ExePath**パラメータ)。

   または、コマンドライン管理ツールの コマンド を使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow LinuxFirewall.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset LinuxFirewall.ExePath <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset LinuxFirewall.ExePath -r
   ```

2. 設定にDr.Web Firewall for Linuxコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、drweb-firewallパッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *Network Checkerを使用することができません。*

エラーコード: x123

説明: Dr.Web Network Checkerコンポーネントがありません(ダウンロードしたファイルのスキャンに必要です)。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認く

ださい（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. **drweb-netcheck**実行ファイルへのパスを確認してください。必要に応じてパスを変更します（設定ファイルの`[Netcheck]`セクションにある**ExePath**パラメータ）。

   または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow Netcheck.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Netcheck.ExePath <new path>
   ```

   パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset Netcheck.ExePath -r
   ```

2. 設定にDr.Web Network Checkerコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、`drweb-netcheck`パッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法についてはDr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合はテクニカルサポートまでご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ： *CloudDを使用することができません。*

エラーコード： `x124`

説明： Dr.Web CloudDが見つかりません（Dr.Web Cloudサービスへの要求に必要です）。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください（デフォルトでは、OSによって `/var/log/syslog`ファイルまたは `/var/log/messages`ファイルにあります）。また、コマンド**drweb-ctl** `log`を使用することもできます。

エラーの解決：

1. **drweb-cloudd**実行ファイルへのパスを確認してください。必要に応じてパスを変更します（設定ファイルの`[CloudD]`セクションにある**ExePath**パラメータ）。

   または、コマンドライン管理ツールのコマンドを使用することもできます。

   現在のパラメータ値を確認するには、以下のコマンドを実行します。

   ```
   $ drweb-ctl cfshow CloudD.ExePath
   ```

   新しいパラメータ値を設定するには、以下のコマンドを実行します。

   ```
   # drweb-ctl cfset CloudD.ExePath <new path>
   ```

パラメータ値をデフォルトに復元するには、以下のコマンドを実行します。

```
# drweb-ctl cfset CloudD.ExePath -r
```

2. 設定にDr.Web CloudDコンポーネントの設定が含まれていない場合、または正しいパスを入力した後もエラーが継続する場合は、drweb-cloudddパッケージをインストールまたは再インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

エラーメッセージ: *予期せぬエラーです。*

エラーコード: x125

説明: いずれかのコンポーネントの動作に、予期せぬエラーが発生しました。

エラーの発生した場所と理由に関する詳細については、Dr.Web for UNIX Internet Gatewaysログをご確認ください(デフォルトでは、OSによって /var/log/syslogファイルまたは /var/log/messagesファイルにあります)。また、コマンドdrweb-ctllogを使用することもできます。

エラーの解決:

1. 以下のコマンドを入力し、Dr.Web for UNIX Internet Gatewaysを再起動します。

   ```
   # service drweb-configd restart
   ```

引き続きエラーが発生する場合は テクニカルサポート までご連絡のうえ、エラーコードをお伝えください。

コードのないエラー

症状:	Dr.Web MailD、SpIDer Gate、Dr.Web ICAPD(表示されるコンポーネントのリストはインストールされている製品によって異なります)はメッセージをスキャンしません。Dr.Web for UNIX Internet Gatewaysログにメッセージ「開いているファイルが多すぎます」が表示されます。
説明:	データスキャンの負荷が大きいため、Dr.Web Network Checkerは利用可能なファイル記述子(ディスクリプタ)数の上限に達しました。

エラーの解決:

1. **ulimit** -nコマンドを使用してアプリケーションで使用可能なオープンファイルの記述子数上限を引き上げます(Dr.Web for UNIX Internet Gatewaysの記述子数のデフォルト上限は16384です)。

   場合によっては、システムサービス**systemd**は指定した上限の変更を無視できます。この場合、ファイル/etc/systemd/system/drweb-configd .service.d/limits.confを編集し(ない場合は作成して)、変更後の上限値を指定します。

   ```
   [Service]
   LimitNOFILE=16384
   ```

systemdの利用可能な上限のリストはドキュメント**man** `systemd.exec`で確認できます。

2. 上限を変更したら、次のコマンドを実行してDr.Web for UNIX Internet Gatewaysを再起動します。

```
# service drweb-configd restart
```

引き続きエラーが発生する場合は、テクニカルサポートまでご連絡ください。

症状:	Webブラウザは Dr.Web管理Webインターフェースへの接続を確立できません。Dr.Webアンチウイルスソリューションのコンポーネントは実行中のプロセスのリスト（**ps** `ax	grep drweb`）に含まれていません。**drweb-ctl** `rawscan`以外の**drweb-ctl** *<command>*を実行しようとすると、次のいずれかのエラーが発生します。  `Error: connect: No such file or directory: "`*<path>*`/.com.drweb.public"`  または  `Error: connect: Connection refused: "`*<path>*`/.com.drweb.public".`
説明:	設定デーモンDr.Web ConfigDが使用できないため、Dr.Web for UNIX Internet Gatewaysを起動できません。

エラーの解決:

1. 次のコマンドを実行します。

```
# service drweb-configd restart
```

   Dr.Web ConfigDとDr.Web for UNIX Internet Gatewaysを再起動します。

2. このコマンドがエラーメッセージを返した場合、または効果がない場合は、`drweb-configd`コンポーネント（パッケージ）を個別にインストールしてください。

   これは**PAM**認証がシステムで使用されていないことを意味する場合があることにも注意してください。その場合は、PAMをインストールして設定します（Dr.Web for UNIX Internet Gatewaysは**PAM**がないと正しく動作できません）。

3. エラーが解決しない場合は、Dr.Web for UNIX Internet Gatewaysを削除してから再度インストールしてください。

   Dr.Web for UNIX Internet Gatewaysまたはそのコンポーネントのインストールとアンインストールの方法については Dr.Web for UNIX Internet Gatewaysをインストールするおよび Dr.Web for UNIX Internet Gatewaysをアンインストールするのセクションを参照してください。

引き続きエラーが発生する場合は、テクニカルサポートまでご連絡ください。

症状:	1. SpIDer Gateを無効にした後、すべてのネットワーク接続が切断され（SSH／FTPプロトコル経由での送信、および場合によっては受信も）、再確立できなくなります。 2. 以下のコマンドを使用して、**NetFilter**（**iptables**）ルール全体を検索する。 `# iptables-save	grep "comment --comment --comment"` を使用すると、空以外の結果が返ってくる。

説明:	このエラーは1.4.15より前のバージョンの不正な**NetFilter**(**iptables**)動作に関連しています。この内部エラーのため、SpIDer Gateが一意のラベル(コメント)が付いたルールをルールのリストに追加すると、そのルールは誤って追加されます。その結果、SpIDer Gateはシャットダウン時に接続の迂回ルールを削除できません。

エラーの解決:

1. SpIDer Gateモニターを再度有効にしてください
2. SpIDer Gateを無効にする必要がある場合、以下のコマンドを使用して、**NetFilter**(**iptables**)の不正なルールを削除してください

```
# iptables-save | grep -v "comment --comment --comment" | iptables-restore
```

iptables-save および **iptables-restore** コマンドにはルート権限が必要です。権限を昇格するには**su** および **sudo** コマンドを使用することができます。また、このコマンドは正しくないコメントを持つすべてのルール(例:同じくトラフィックのリダイレクトを実行する他のアプリケーションによって追加されたものなど)を削除するという点に注意してください。

追加情報:

- この問題の発生を防ぐため、お使いのOSをアップグレードすることが推奨されます(または、少なくとも**NetFilter** をバージョン1.4.15以降に)。
- また、**iptables**ユーティリティを使用して必要なルールを指定することで接続を手動でSpIDer Gateへリダイレクトさせる場合は、Dr.Web Firewallの設定で、SpIDer Gateへの接続のリダイレクトを手動モードに切り替えることができます(この方法は推奨されません)。
- 詳細はマニュアル**man:** `drweb-firewall(1)`、`drweb-gated(1)`、`iptables(8)`を参照してください。

引き続きエラーが発生する場合は、テクニカルサポート までご連絡ください。

エラーの内部カタログ

エラーコード	シンボリック文字・記号	エラーについての内部メッセージ	説明
0	EC_OK	*成功*	これはエラーではありません
1	EC_MONITOR_IPC_ERROR	*モニターチャネルに関するエラー*	エラーx1
2	EC_ALREADY_IN_PROGRESS	*操作はすでに実行中です*	エラーx2
3	EC_IN_PENDING_STATE	*操作は保留中です*	エラーx3
4	EC_INTERRUPTED_BY_USER	*ユーザーによって中断されました*	エラーx4
5	EC_CANCELED	*操作がキャンセルされました*	エラーx5
6	EC_LINK_DISCONNECTED	*リンクが切断されました*	エラーx6

エラーコード	シンボリック文字・記号	エラーについての内部メッセージ	説明
7	EC_BAD_MESSAGE_SIZE	*無効なIPCメッセージサイズです*	エラーx7
8	EC_BAD_MESSAGE_FORMAT	*無効なIPCメッセージフォーマットです*	エラーx8
9	EC_NOT_READY	*準備が完了していません*	エラーx9
10	EC_NOT_INSTALLED	*コンポーネントがインストールされていません*	エラーx10
11	EC_UNEXPECTED_MESSAGE	*予期せぬIPCメッセージです*	エラーx11
12	EC_PROTOCOL_VIOLATION	*プロトコル違反です*	エラーx12
13	EC_UNKNOWN_STATE	*サブシステムの状態が未知です*	エラーx13
20	EC_NOT_ABSOLUTE_PATH	*パスは絶対パスでなければなりません*	エラーx20
21	EC_NO_MEMORY	*十分なメモリがありません*	エラーx21
22	EC_IO_ERROR	*IOエラー*	エラーx22
23	EC_NO_SUCH_ENTRY	*そのようなファイルまたはディレクトリがありません*	エラーx23
24	EC_PERMISSION_DENIED	*パーミッションが拒否されました*	エラーx24
25	EC_NOT_A_DIRECTORY	*ディレクトリではありません*	エラーx25
26	EC_DATA_CORRUPTED	*データファイルが破損しています*	エラーx26
27	EC_FILE_EXISTS	*ファイルはすでに存在しています*	エラーx27
28	EC_READ_ONLY_FS	*読み取り専用ファイルシステム*	エラーx28
29	EC_NETWORK_ERROR	*ネットワークエラー*	エラーx29
30	EC_NOT_A_DRIVE	*ドライブではありません*	エラーx30
31	EC_UNEXPECTED_EOF	*予期せぬEOFがあります*	エラーx31
32	EC_FILE_WAS_CHANGED	*ファイルが変更されています*	エラーx32
33	EC_NOT_A_REGULAR_FILE	*通常ファイルではありません*	エラーx33
34	EC_NAME_ALREADY_IN_USE	*名前はすでに使用されています*	エラーx34
35	EC_HOST_OFFLINE	*ホストがオフラインです*	エラーx35
36	EC_LIMIT_REACHED	*リソースの上限に達しています*	エラーx36
37	EC_CROSS_DEVICE_LINK	*マウントポイントが異なります*	エラーx37

エラーコード	シンボリック文字・記号	エラーについての内部メッセージ	説明
38	EC_UNPACKING_ERROR	*アンパックエラー*	エラーx38
40	EC_BASE_CORRUPTED	*ウイルスデータベースが破損しています*	エラーx40
41	EC_OLD_BASE_VERSION	*サポートされていないバージョンのウイルスデータベースです*	エラーx41
42	EC_EMPTY_BASE	*ウイルスデータベースが空です*	エラーx42
43	EC_CAN_NOT_BE_CURED	*オブジェクトを修復できません*	エラーx43
44	EC_INVALID_BASE_SET	*サポートされていないウイルスデータベースの組み合わせです*	エラーx44
45	EC_SCAN_LIMIT_REACHED	*スキャンの上限に達しています*	エラーx45
47	EC_AUTH_FAILED	*認証に失敗しました*	エラーx47
48	EC_NOT_AUTHORIZED	*認証に失敗しました*	エラーx48
49	EC_INVALID_TOKEN	*無効なアクセストークンです*	エラーx49
60	EC_INVALID_ARGUMENT	*無効な引数です*	エラーx60
61	EC_INVALID_OPERATION	*無効な操作です*	エラーx61
62	EC_ROOT_ONLY	*スーパーユーザー権限が必要です*	エラーx62
63	EC_STANDALONE_MODE_ONLY	*集中管理モードでは許可されていません。*	エラーx63
64	EC_NON_SUPPORTED_OS	*サポートされていないOSです*	エラーx64
65	EC_NOT_IMPLEMENTED	*実装されていない機能です*	エラーx65
66	EC_UNKNOWN_OPTION	*未知のオプションです*	エラーx66
67	EC_UNKNOWN_SECTION	*未知のセクションです*	エラーx67
68	EC_INVALID_OPTION_VALUE	*無効なオプション値です*	エラーx68
69	EC_INVALID_STATE	*無効な状態です*	エラーx69
70	EC_NOT_LIST_OPTION	*使用可能な値は1つのみです*	エラーx70
71	EC_INVALID_TAG	*無効なタグ値です*	エラーx71
80	EC_RECORD_NOT_FOUND	*レコードが見つかりません*	エラーx80
81	EC_RECORD_BUSY	*レコードは現在処理中です*	エラーx81

エラーコード	シンボリック文字・記号	エラーについての内部メッセージ	説明
82	EC_QUARANTINED_FILE	*ファイルはすでに隔離済みです*	エラーx82
89	EC_BACKUP_FAILED	*更新前にバックアップを行うことができません*	エラーx89
90	EC_BAD_DRL_FILE	*無効なDRLファイルです*	エラーx90
91	EC_BAD_LST_FILE	*無効なLSTファイルです*	エラーx91
92	EC_BAD_LZMA_FILE	*無効な圧縮ファイルです*	エラーx92
93	EC_PROXY_AUTH_ERROR	*プロキシ認証エラーです*	エラーx93
94	EC_NO_UPDATE_SERVERS	*使用可能な更新サーバーがありません*	エラーx94
95	EC_BAD_KEY_FORMAT	*キーファイルのフォーマットが無効です*	エラーx95
96	EC_EXPIRED_KEY	*ライセンスは有効期限が切れています*	エラーx96
97	EC_NETWORK_TIMEDOUT	*ネットワークオペレーションのタイムアウト*	エラーx97
98	EC_BAD_CHECKSUM	*無効なチェックサムです*	エラーx98
99	EC_BAD_TRIAL_KEY	*無効なトライアルライセンス*	エラーx99
100	EC_BLOCKED_LICENSE	*ブロックされているライセンスキーです*	エラーx100
101	EC_BAD_LICENSE	*無効なライセンスです*	エラーx101
102	EC_BAD_CONFIG	*無効な設定です*	エラーx102
104	EC_BAD_EXECUTABLE	*無効な実行ファイルです*	エラーx104
105	EC_NO_CORE_ENGINE	*コアエンジンは使用できません*	エラーx105
106	EC_NO_VIRUS_BASES	*ウイルスデータベースがありません*	エラーx106
107	EC_APP_TERMINATED	*プロセスはシグナルによって中断されました*	エラーx107
108	EC_APP_CRASHED	*予期せぬプロセスの中断です*	エラーx108
109	EC_INCOMPATIBLE	*互換性のないソフトウェアが検出されました*	エラーx109
112	EC_NO_DWS_BASES	*Webリソースデータベースがありません*	エラーx112
115	EC_NO_LOOKUPD	*LookupDは使用できません*	エラーx115
117	EC_NO_GATED	*GateDは使用できません*	エラーx117
119	EC_NO_SCAN_ENGINE	*ScanEngineは使用できません*	エラーx119

エラーコード	シンボリック文字・記号	エラーについての内部メッセージ	説明
120	EC_NO_FILE_CHECK	*FileCheckは使用できません*	エラーx120
121	EC_NO_ESAGENT	*ESAgentは使用できません*	エラーx121
122	EC_NO_FIREWALL	*Firewallは使用できません*	エラーx122
123	EC_NO_NET_CHECK	*NetCheckは使用できません*	エラーx123
124	EC_NO_CLOUDD	*CloudDを使用できません*	エラーx124
125	EC_UNEXPECTED_ERROR	*予期せぬエラーです*	エラーx125

付録G. 略語のリスト

このマニュアルでは、以下の用語は説明なしに使用されます。

文字・記号	説明
AD	Microsoft Active Directory
DN	（LDAP）Distinguished Name（識別名）
EPM	ESP Package Manager（ESPパッケージマネージャー）
FQDN	Fully Qualified Domain Name（完全修飾ドメイン名）
GID	Group ID（システムユーザーグループID）
GNU	GNUプロジェクト（GNU is Not Unix）
HTML	HyperText Markup Language（ハイパーテキストマークアップ言語）
HTTP	HyperText Transfer Protocol（ハイパーテキスト転送プロトコル）
HTTPS	HyperText Transfer Protocol Secure（SSL/TLS経由）
ICAP	Internet Content Adaptation Protocol（インターネットコンテンツアダプテーションプロトコル）
ID	ID（識別子）
IP	Internet Protocol（インターネットプロトコル）
LDAP	Lightweight Directory Access Protocol（ライトウェイトディレクトリ アクセスプロトコル）
MBR	Master Boot Record（マスターブートレコード）
OID	（SNMP）Object ID（オブジェクトID）

文字・記号	説明
PID	Process ID（システムプロセスID）
PAM	Pluggable Authentication Modules（プラガブル認証モジュール）
RPM	Red Hat Package Manager（Red Hatパッケージマネージャー）
RRA	Round-Robin Archive（ラウンドロビンアーカイブ）
RRD	Round-Robin Database（ラウンドロビンデータベース）
SNI	Server Name Indication（サーバー名表示）
SNMP	Simple Network Management Protocol（シンプルネットワークマネージメントプロトコル）
SP	Service Pack（サービスパック）
SSH	Secure Shell（セキュアシェル）
SSL	Secure Sockets Layer（セキュアソケットレイヤー）
TCP	Transmission Control Protocol（伝送制御プロトコル）
TLS	Transport Layer Security（トランスポート層セキュリティ）
UID	User ID（システムユーザーID）
URI	Uniform Resource Identifier（ユニフォームリソースアイデンティファイア）
URL	Uniform Resource Locator（ユニフォームリソースロケータ）
VBR	Volume Boot Record（ボリュームブートレコード）
OS	Operating System（オペレーティングシステム）
FS	File System（ファイルシステム）