Dr.Web Mail Security Suite
Ver.11.1 簡易構築ガイド
-Linux 用-

株式会社 Doctor Web Pacific

初版 ： 2019/03/27
改訂 ： 2020/09/07

目次

この度は、株式会社 DoctorWebPacific の製品をご購入いただき、誠にありがとうございます。本ガイドは、初めて弊社製品をご利用いただくお客様向けに、Dr.Web Mail Security Suite(以下 MSS)を簡潔に構築いただくための手順を説明する資料となります。なお、詳細な機能や操作の説明に関しましては、製品マニュアルをご参照ください。

1. はじめに

1.1 ライセンス証書の受領

ライセンス証書は、Doctor Web Pacific(以下、DWP)または、DWP パートナー企業より、電子メールか郵送もしくはその両方の方法で、お客様へ送付いたします。

1.2 ライセンス証書に含まれる内容

ライセンス証書には、以下のライセンスに関する情報が記載されておりますので、大切に保管してください。

- custmer(お客様情報)
- product(購入製品名)
- serial number(製品用キーコード)
- license term(ライセンス期間)
- protected objects (購入ライセンス数)

1.3 システム要件

システム要件につきましては、下記 URL をご参照ください。

https://download.geo.drweb.com/pub/drweb/unix/mail/11.1/documentation/html/en/dw_9_sysrequirements.htm

2. 環境前提条件

本書は、下記の環境で動作確認の上作成しております。

- OS
  Cent OS 7.5 (64bit)
- MTA およびバージョン
  Postfix 2.10.1
- selinux
  無効
- firewalld
  無効

3. 準備

3.1 インストール環境の確認

3.1.1 インストール済みパッケージの確認

OS 毎に以下のパッケージがインストールされているか確認し、インストールされていない場合はインストールしてください。

- Cent OS 7.5

  glibc.i686、glibc.x86_64、glibc-common.x86_64、nss-softokn-freebl.i686、nss-softokn-freebl.x86_64、perl、perl-Data-Dumper、perl-Sys-Syslog

3.1.2 以前のバージョンの MSS がインストールされている場合

インストールするバージョンよりも古い MSS がインストールされている場合は、「6.8 以前のバージョンの MSS のアンインストール」に記載の手順に従い、事前にアンインストールを実施してください。

3.2 リポジトリ設定

MSS をリポジトリからインストールする場合、以下のコマンドを実行してください。

```
# wget https://repo.drweb.com/drweb/drweb-repo11.1.rpm
# rpm –ivh drweb-repo11.1.rpm
```

3.3 ファイル

以下のファイルを用意してください。キーファイルおよびインストーラの入手方法については、「Dr.Web ダウンロード＆アクティベーションガイド」を参照してください。

尚、MSS をリポジトリからインストールする場合は、インストーラ(.run ファイル)のダウンロードは不要です。

- キーファイル等

  drweb32.key もしくは agent.key を用意し、インストール対象のサーバにコピーしてください。

  ESS サーバ(バージョン 11)の Agent として接続する場合は、当該サーバの drwcsd.pub ファイルを用意してください。

  ※ AV DESK サーバの Agent として接続することはできません。

- インストーラ

  インストーラ(.run ファイル)を用意し、インストール対象のサーバにコピーしてください。

4. インストール

4.1 リポジトリからのインストール

1) 以下のコマンドを実行し、MSS のインストールを実行します。

```
# yum install drweb-mail-servers
```

2) インストールが完了した後、キーファイル(drweb32.key もしくは agent.key)を/etc/opt/drweb.com/に drweb32.key としてコピーします。

3) 以下のコマンドを実行し、サービスを再起動します。

```
# systemctl restart drweb-configd
```

4) 以下のコマンドを実行し、ライセンスが正常に読み込まれているか確認します。

```
# drweb-ctl license
License number <Key No.>, expires <ライセンス期限> (<残り日数>)
```

4.2 インストーラ(.run)からのインストール

1) インストーラ(.run ファイル)のパーミッションを変更し、実行権を付与します。

```
# chmod +x <インストーラ名>
```

2) 以下のコマンドを実行します。

```
# ./<インストーラ名>
```

※ ファイルの解凍が始まります。

3) 以下のメッセージが表示されたら、「yes」と入力し、「Enter」キーを押します。

```
This installation script will help you install Dr.Web for Mail Servers

Do you want to continue? (YES/no)
```

4) 以下のメッセージが表示されたら、「yes」と入力し、「Enter」キーを押します。

```
Do you agree with the terms of this license? (yes/NO)
```

5) 以下のメッセージが表示されたら、「Enter」キーを押します。

```
Dr.Web packages repository is added to your repositories list.
```

※ ESS サーバと接続させる場合は以降の手順は行わず、「5.5 ESS サーバとの接続」を参照してください。

6) キーファイル(drweb32.key もしくは agent.key)を/etc/opt/drweb.com/drweb32.key としてコピーします。

7) 以下のコマンドを実行し、サービスを再起動します。

```
# systemctl restart drweb-configd
```

8) 以下のコマンドを実行し、ライセンスが正常に読み込まれているか確認します。

```
# drweb-ctl license
License number <Key No.>, expires <ライセンス期限> (<残り日数>)
```

9) 以下のコマンドを実行し、MSS のコンポーネント(プログラム)を更新します。

```
# yum update drweb*
```

5. MTA との連携

MSS11 では、フィルターモードとプロキシモードの 2 つの連携モードがあります。使用されている OS および MTA により、利用可能なモードが異なりますのでご注意ください。

また、連携設定を行う前に MTA を使用して正常にメールの送受信ができることを確認してください。

- フィルターモード

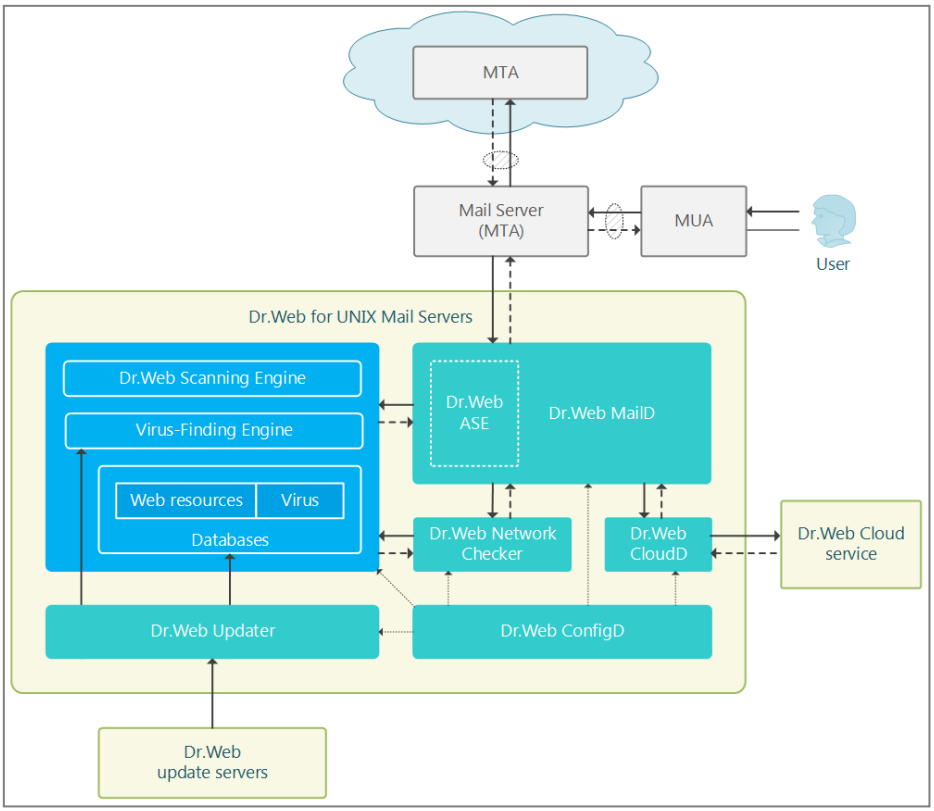
MTA として、Postfix、Sendmail、Exim のいずれかを使用している場合、フィルターモードを使用してください。

※ FreeBSD ではこちらのモードを使用してください。

※ MSS6 における drweb-mail-servers と同様に、MTA の設定変更が必要となります。

※ 本ガイドでは、こちらについて記載しております。

<<フィルターモードにおけるメールの流れ>>



➢ プロキシモード

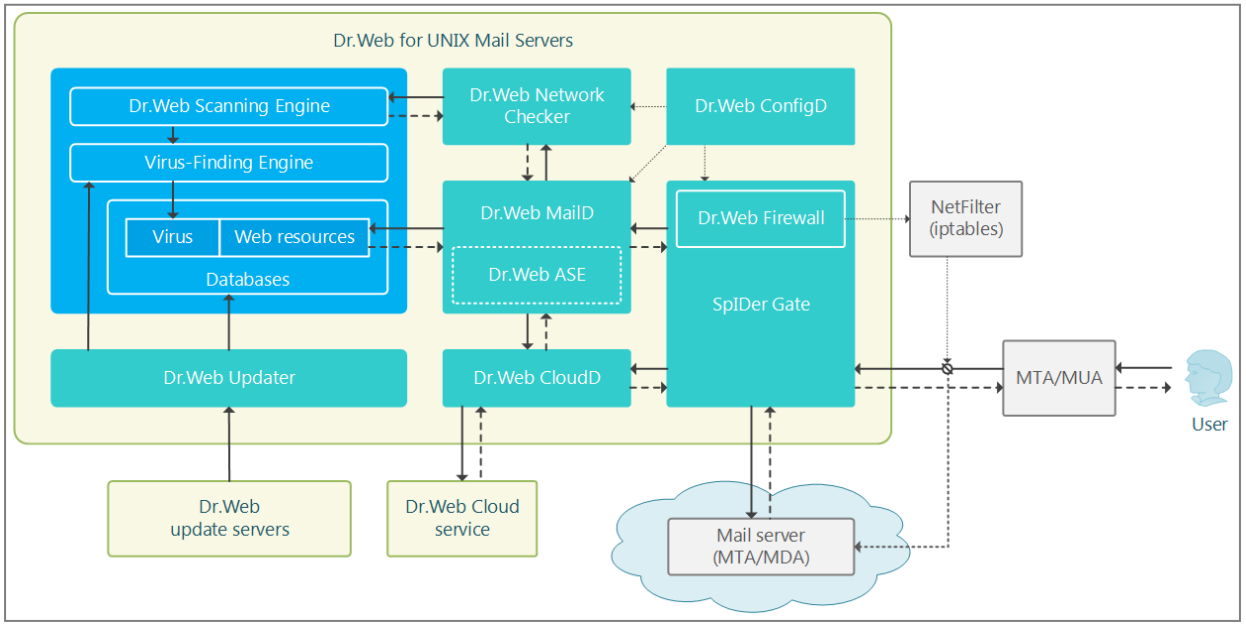
Postfix、Sendmail、Exim 以外の MTA を使用している場合には、フィルターモードではなく、MSS11 を透過型プロキシとして動作させるプロキシモードを使用してください。

※ FreeBSD では、使用できません。

※ MSS6 における drweb-mail-gateways と異なり、メールの転送機能はありません。

※ 本ガイドでは、記載しておりません。

<<プロキシモードにおけるメールの流れ>>



5.1. 設定方法

MSS11 の設定は、”drweb-ctl”コマンドもしくは Web インターフェースから実施することができます。

初期状態では、MSS11 をインストールしたサーバ上で以下の URL にアクセスすることにより、Web インターフェースを開くことができます。

URL　　：　https://127.0.0.1:4443/

ID　　：　root

Password ：　root のパスワード

※　初期状態では、他の端末から Web インターフェースを開くことはできません。

※　他の端末から Web インターフェースへのアクセスを可能にする場合は、以下のコマンドを実行してください。

```
# drweb-ctl cfset HTTPD. AdminListen <IP アドレス>:4443
```

5.1.1　フィルターモードでのメール処理に関する初期設定

メールの処理は、フィルターモードでは"MailD.MilterHook"内で Lua スクリプトを用いて設定します。初期設定では、以下の内容が設定されています。

尚、MSS6 と異なり、脅威のみを削除してメールを配信することや、メール自体を隔離することはできません。

また、ご利用中のライセンスにアンチスパムが含まれていない場合は、スパムに関する処理を指定している箇所は削除してください。

- Spam Score が 100 以上のメールを拒否(reject)し、それ以外は以降の処理を実施した後に配信。
  ※　以下は Spam Score が 100 未満のメールに対する処理となります。
- "X-DrWeb-SpamScore"、"X-DrWeb-SpamState"、"X-DrWeb-SpamDetail"、"X-DrWeb-SpamVersion"、"X-AntiVirus"の各ヘッダーを追加。
- メールから"known_virus", "virus_modification", "unknown_virus", "adware", "dialer"が検出された場合は、repack(元のメールを quarantine.zip に圧縮)。
- メールに指定されたカテゴリ("infection_source", "not_recommended", "owners_notice")に該当する URL が含まれる場合は、repack(元のメールを quarantine.zip に圧縮)。

```
local dw = require "drweb"

function milter_hook(ctx)
	-- Reject the message if it is likely spam
	if ctx.message.spam.score >= 100 then
		dw.notice("Spam score: " .. ctx.message.spam.score)
		return {action = "reject"}
	else
		-- Assign X-Drweb-Spam headers in accordance with spam report
		ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
		ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
		ctx.modifier.add_header_field("X-DrWeb-SpamDetail", ctx.message.spam.reason)
		ctx.modifier.add_header_field("X-DrWeb-SpamVersion", ctx.message.spam.version)
	end
	-- Check if the message contains viruses, repack if so
	for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer"}} do
		ctx.modifier.repack()
		dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
	end
	-- Repack if unwanted URL has been found
	for url in ctx.message.urls{category = {"infection_source", "not_recommended",
"owners_notice"}} do
		ctx.modifier.repack()
		dw.notice("URL found: " .. url .. "(" .. url.categories[1] .. ")")
	end
	-- Assign X-AntiVirus header
	ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: ]")
	-- Accept the message with all scheduled transformations applied

	return {action = 'accept'}

end
```

5.1.2 プロキシモードでのメール処理に関する初期設定

メールの処理は、プロキシモードでは"LinuxFirewall.RuleSet"内で処理ルールを設定します。初期設定では、以下の内容(smtp、imap、pop3 に関するもののみを記載)が設定されています。

尚、MSS6 と異なり、脅威のみを削除してメールを配信することや、メール自体を隔離することはできません。

- 脅威のカテゴリが Dr.Web Firewall for Linux の File Filter 設定でブロック対象となっているものは、SMTP の場合は拒否(REJECT)、POP3 および IMAP の場合は圧縮して配信(REPACK as _match)
- メール内の URL のカテゴリが Dr.Web Firewall for Linux の Web Filter 設定でブロック対象となっているものは、SMTP の場合は拒否(REJECT)、POP3 および IMAP の場合は圧縮して配信(REPACK as _match)

LinuxFirewall.RuleSet1 = ：set UnwrapSSL = false
LinuxFirewall.RuleSet1 = divert output：set HttpTemplatesDir = "output"
LinuxFirewall.RuleSet1 = divert input：set HttpTemplatesDir = "input"
LinuxFirewall.RuleSet1 = divert forward：set HttpTemplatesDir = "output"
LinuxFirewall.RuleSet1 = ：set MailTemplatesDir = "firewall"

LinuxFirewall.RuleSet5 = protocol in (Http), direction request, url_host in "LinuxFirewall.Blacklist"：BLOCK as BlackList
LinuxFirewall.RuleSet5 = protocol in (Http), direction request, url_host in "LinuxFirewall.Whitelist"：PASS

LinuxFirewall.RuleSet7 = protocol in (Http), direction request, url_category in "LinuxFirewall.BlockCategory"：BLOCK as _match

LinuxFirewall.RuleSet9 = protocol in (Http), divert input, direction request, threat_category in "LinuxFirewall.BlockThreat"：BLOCK as _match
LinuxFirewall.RuleSet9 = protocol in (Http), direction response, threat_category in "LinuxFirewall.BlockThreat"：BLOCK as _match
LinuxFirewall.RuleSet9 = protocol in (Smtp), threat_category in "LinuxFirewall.BlockThreat"：REJECT
LinuxFirewall.RuleSet9 = protocol in (Smtp), url_category in "LinuxFirewall.BlockCategory"：REJECT
LinuxFirewall.RuleSet9 = protocol in (Pop3, Imap), threat_category in "LinuxFirewall.BlockThreat"：REPACK as _match
LinuxFirewall.RuleSet9 = protocol in (Pop3, Imap), url_category in "LinuxFirewall.BlockCategory"：REPACK as _match

5.2. フィルターモード

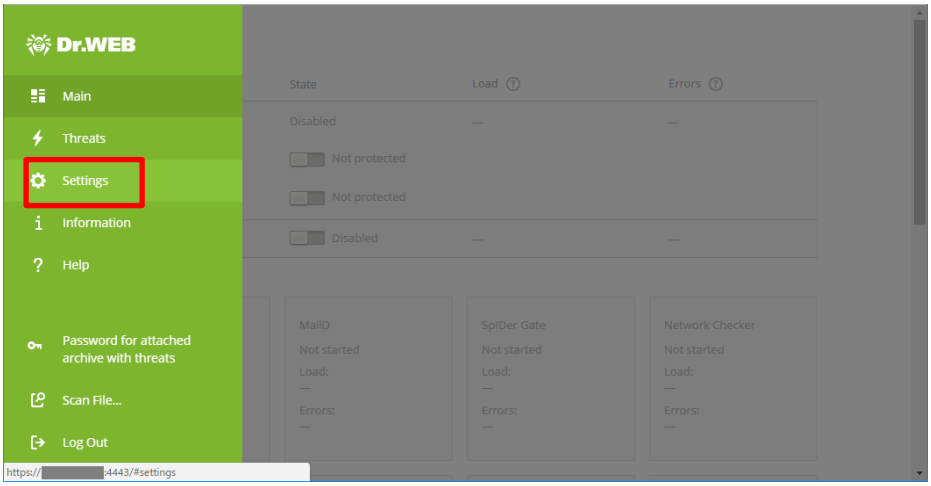
5.2.1 MTA との連携設定(MSS11 の設定)

MTA と連携するための Socket の設定を行います。

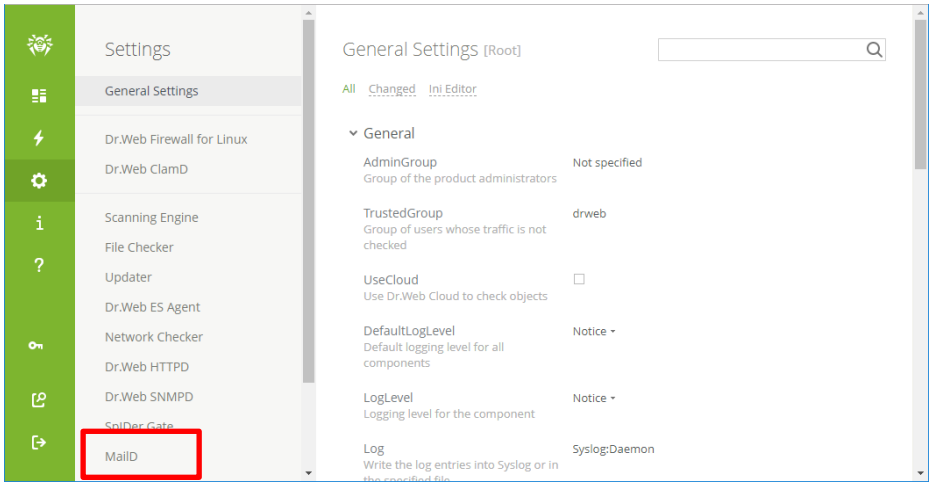
※ 以降は MTA として postfix を使用している場合の設定となります。

1) Web インターフェースにログインします。

2) 左側のメニューから[Settings]をクリックします。

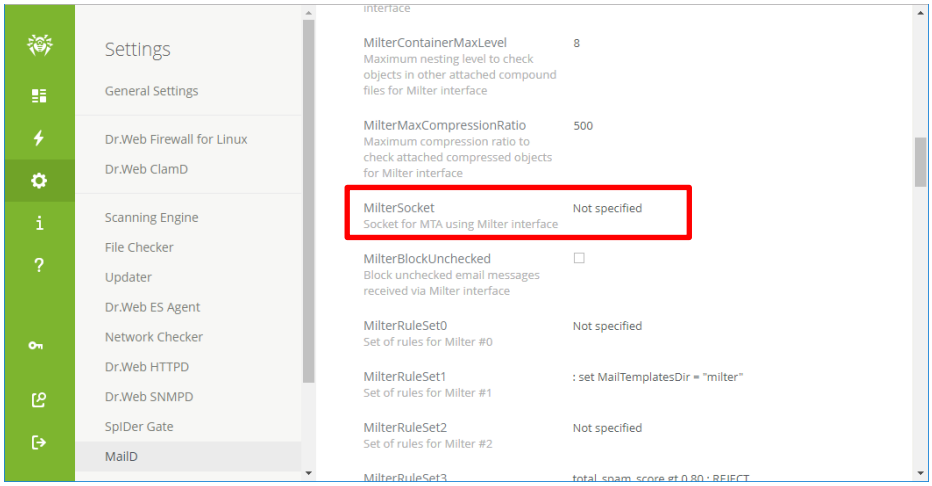


3) 次に「MailD」をクリックします。



4) 画面右側に表示された「MailD」ページ内の「Milter Connections」セクションに移動します。

5) 「MilterScoket」の”Not specified”と書かれている箇所をクリックします。



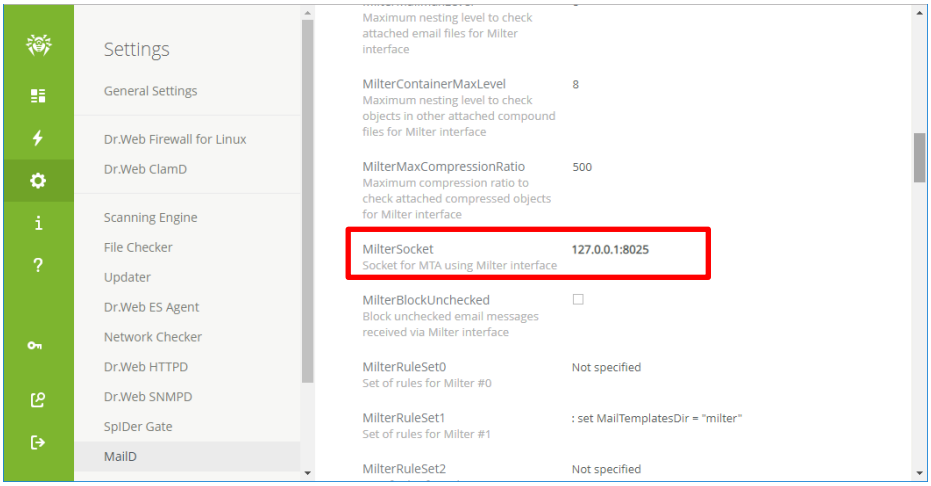
6) 表示された画面で、MTA が動作するサーバの IP アドレスと連携用のポートを”**<IP アドレス>:<ポート番号>**“の形式で指定し、「Save」ボタンをクリックします。

※ 上記は、MTA と MSS11 が同一サーバー上にインストールされている場合の指定例となり、MilterSocket として「127.0.0.1:8025」を指定しています。

※　“drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset MailD.MilterSocket 127.0.0.1:8025
```

7)　MilterSocket が設定されたことを確認します。



※　“drweb-ctl”コマンドで確認する場合は、下記となります。

```
# drweb-ctl cfshow | grep MailD.MilterSocket
```

5.2.2　MTA との連携設定(Postfix の設定)

MSS11 と連携するための設定を行います。

※　下記マニュアルの内容もご確認ください。

https://download.geo.drweb.com/pub/drweb/unix/mail/11.0/documentation/html/en/dw_9_mta_integration.htm

1)　MTA(Postfix)がインストールされたサーバにログインします。

2)　main.cf を開き、以下の内容を追加します。

```
smtpd_milters = <type>:<MailD socket>
milter_content_timeout = 300s
milter_default_action = tempfail
milter_protocol = 2
```

※　”<type>:<MailD socket>”には、5.2.1 で設定した MilterSocket を”**inet:<IP アドレス>:<ポート番号>**”の形式で指定してください。例えば、MSS11 で MilterSocket として「127.0.0.1:8025」を指定した場合には、main.cf では「inet:127.0.0.1:8025」と指定してください。

3)　Postfix を再起動します。

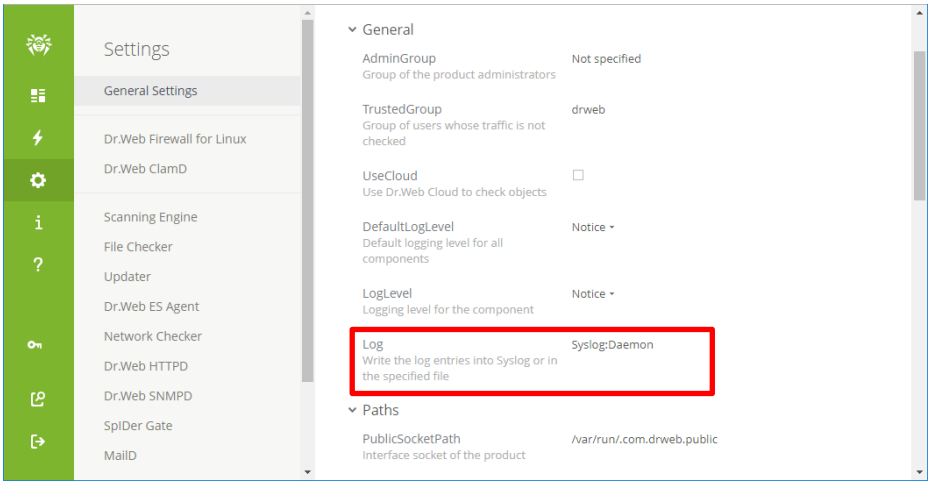
5.2.3　動作確認

正常に連携できているか確認を行います。

初期状態ではログに出力される内容が少ないためログの出力先等を変更後に、PC 上のメールクライアントからメールを送信して動作の確認を行ないます。

※　ログの出力先やログレベルを変更したままの状態で運用される場合、ログはローテートされませんので別途 OS 側でログのローテートの設定が必要です。

1) Web インターフェースにログインします。
2) 左側のメニューから[Settings]をクリックします。
3) 次に「General Settings」をクリックします。
4) 画面右側に表示された「General Settings」ページ内の「General」セクションに移動します。
5) 「Log」の”Syslog:Daemon”と書かれている箇所をクリックします。

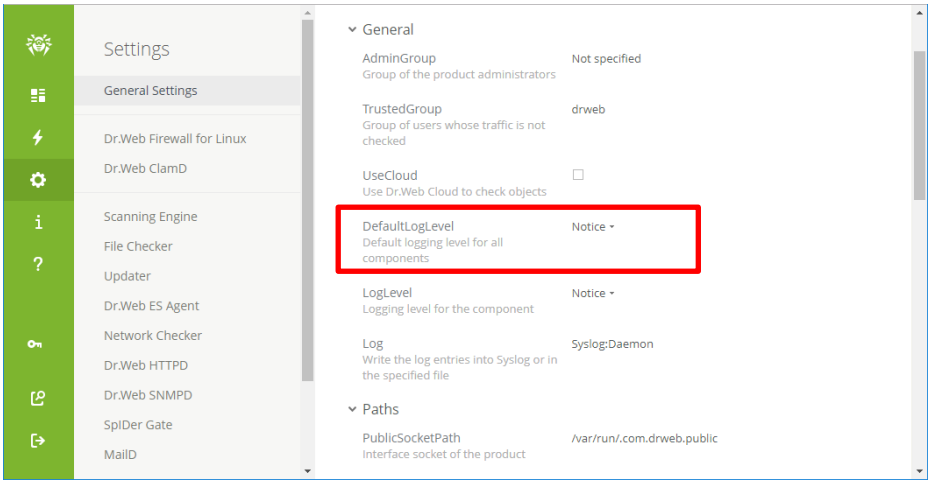


6) “Syslog:Daemon”を”/var/log/drweb.log”に変更し、「Save」をクリックします。

※　“drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset Root.Log /var/log/drweb.log
```

7) 「DefaultLogLevel」の”Notice”と書かれている箇所をクリックし、表示された一覧から”Info”を選択します。



※ “drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset Root.DefaultLogLevel Info
```

8) 「Log」の設定値が”/var/log/drweb.log”に変更されたこと、「DefaultLogLevel」の設定値が”Info”に変更されたことを確認します。

※ “drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfshow | grep Root.Log
# drweb-ctl cfshow | grep Root.DefaultLogLevel
```

9) PC 上のメールクライアントから、当該サーバ上のメールアドレス宛にメールを送信します。

10) “/var/log/drweb.log”に以下のようなログが出力されたことを確認します。

```
2018-Aug-23 13:54:15 [SE-1930] F-3030: Info: Scan "/tmp/com.drweb.ncheck/463e-fd51-6290-ff26"
```

※ 初期状態では、メールのヘッダーに MSS11 でスキャンした事を示すものは追加されません。

11) 変更した「Log」と「DefaultLogLevel」の設定を元の状態に戻します。

※ 元の状態に戻さない場合には、必ず OS 側でログのローテートの設定を行なってください。

※ “drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset Root.Log Syslog:Daemon
# drweb-ctl cfset Root.DefaultLogLevel Notice
```

6. ケーススタディ

6.1. コマンドを用いた設定の確認と変更

6.1.1 設定の確認

コマンドラインから以下のコマンドを実行すると、現在の設定が出力されます。

```
# drweb-ctl cfshow
```

6.1.2 設定の変更

コマンドラインから以下のコマンドを実行することにより、設定を変更できます。

```
# drweb-ctl cfset <section>.<parameter> <設定値>
```

6.2. Web インターフェース

Web インターフェースを使用することにより、ステータスの確認、設定の変更、ライセンスの更新を行なうことができます。

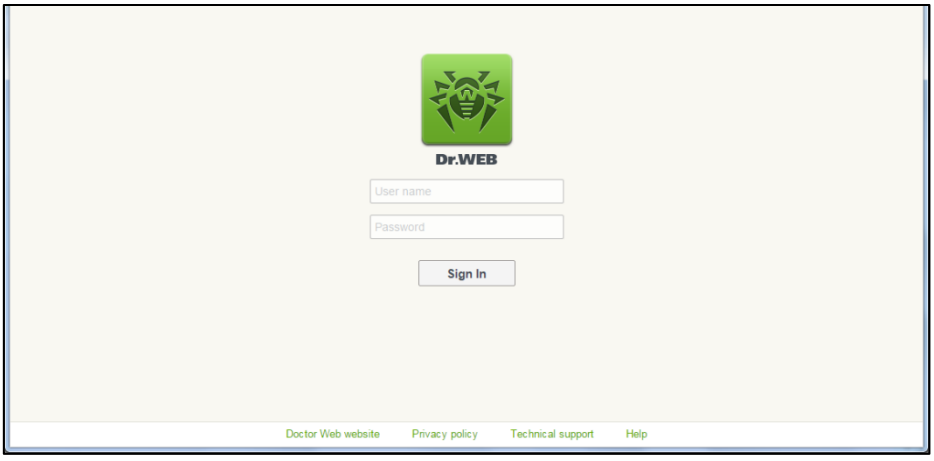
初期状態では、MSS をインストールしたサーバ上で以下の URL にアクセスすることにより、Web インターフェースを開くことができます。

URL : https://127.0.0.1:4443/

ID : root

Password : root のパスワード

※ 初期状態では、他の端末から Web インターフェースを開くことはできません。



※ 他の端末から Web インターフェースへのアクセスを可能にする場合は、以下のコマンドを実行してください。

```
# drweb-ctl cfset HTTPD.AdminListen <IP アドレス>:4443
```

6.3. ライセンス更新

6.3.1　コマンドラインからの更新

1) 新しいライセンスキー(drweb32.key もしくは agent.key)を/etc/opt/drweb.com/に drweb32.key としてコピーします。

2) 以下のサービスを再起動します。

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

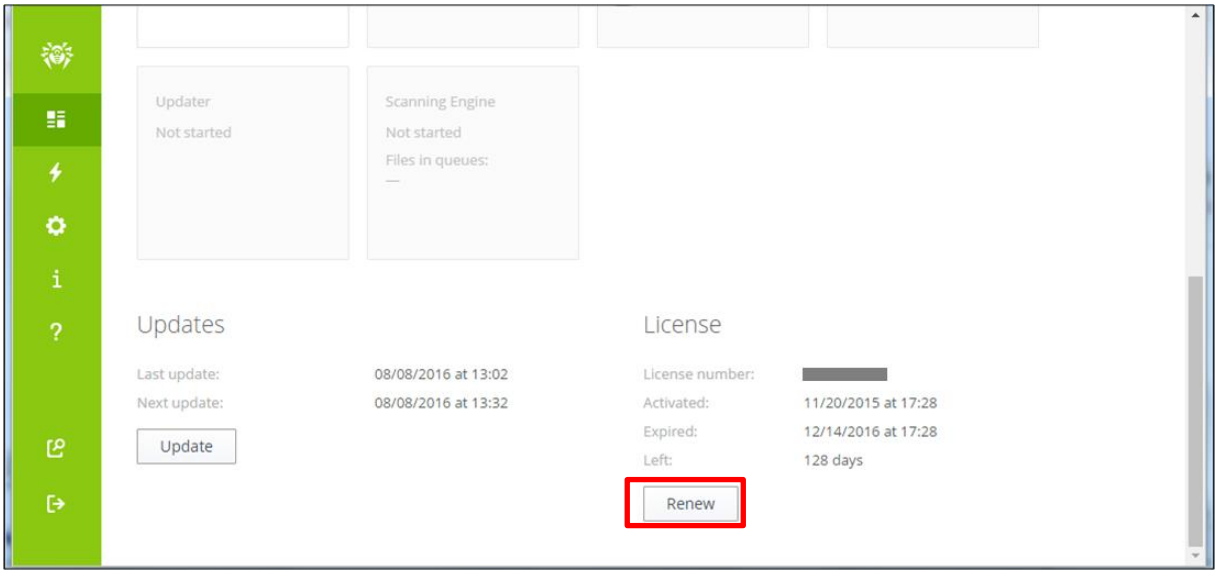
3) 以下のコマンドを実行して、新しいライセンスに更新されたことを確認します。

```
# drweb-ctl license
License number <Key No.>, expires <ライセンス期限> (<残り日数>)
```

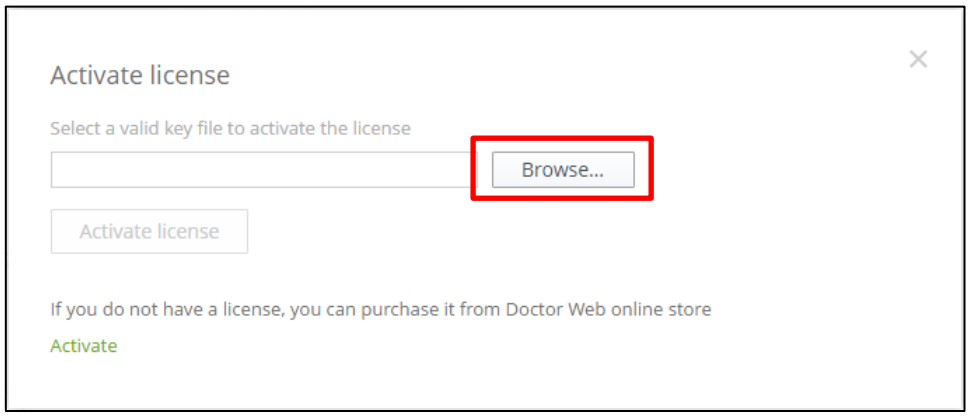
6.3.2　Web インターフェースからの更新

1) Web インターフェースにログインします。

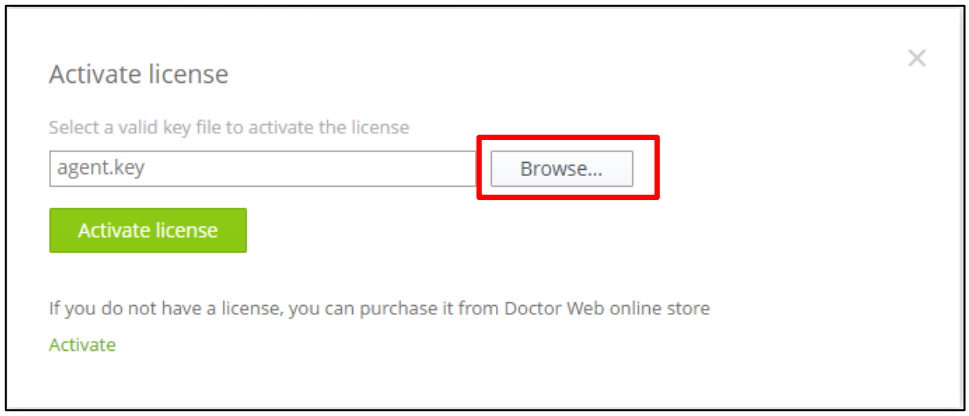
2) [Main]メニュー内の License セクションの「Upload」ボタンをクリックします。



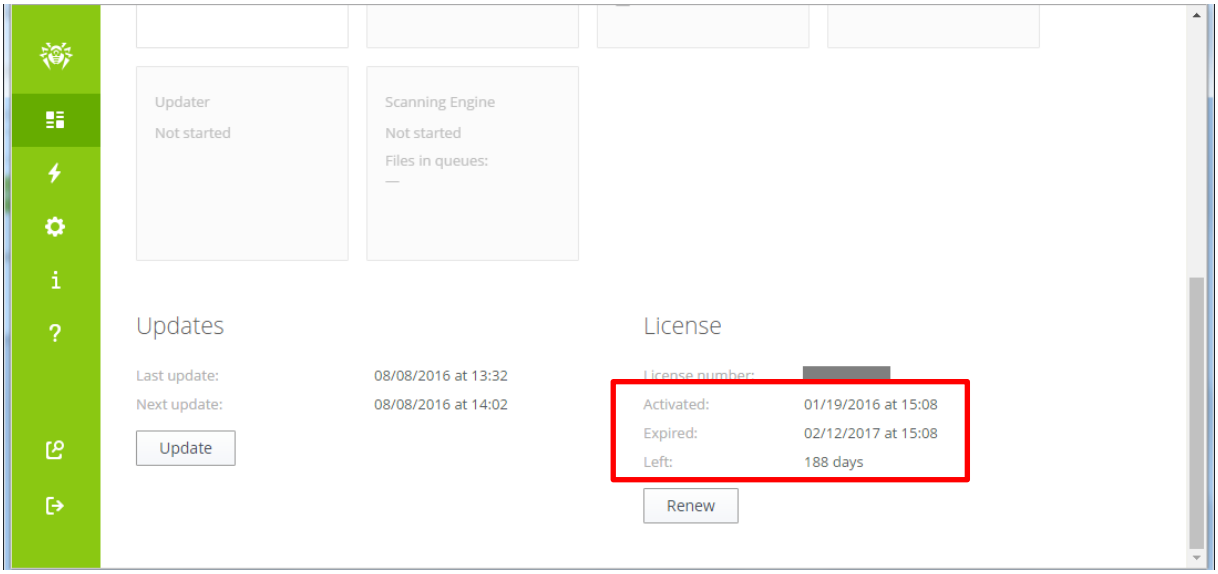
3) 「Browse」ボタンをクリックし、新しいライセンスキー(drweb32.key もしくは agent.key)を指定します。



4) 「Activate license」ボタンをクリックします。



5) Web インターフェース上のライセンス情報が更新されたことを確認します。



6) 以下のコマンドを実行して、新しいライセンスに更新されたことを確認します。

```
# drweb-ctl license
License number <Key No.>, expires <ライセンス期限> (<残り日数>)
```

6.4. MSS のコンポーネントの更新

MSS のコンポーネント(プログラム)は自動では更新されませんが、以下のコマンドを実行すると更新が可能です。

```
# yum update drweb*
```

※ 上記は、Cent OS や RedHat の場合の例です。他の OS やディストリビューションについては、マニュアルをご確認ください。

6.5. 定義ファイルの更新

定義ファイルは、初期設定では 30 分間隔で自動更新されます。手動で更新する場合や更新間隔を変更する場合は、以下の手順にて実施できます。

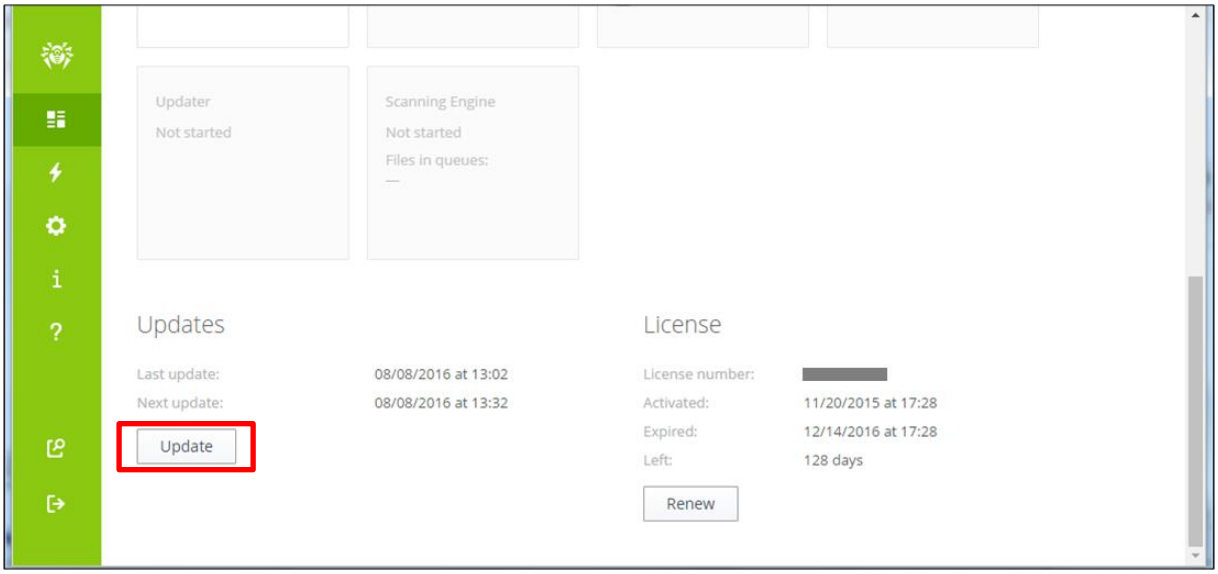
1) 手動更新

- コマンドラインから実行する場合

```
# drweb-ctl update
```

- Web インターフェースから実行する場合

[Main]メニュー内の Updates セクションの「Update」ボタンをクリックします。



2) 更新間隔の変更

- コマンドラインから実行する場合

  下記は更新間隔を”60 分”に変更する場合の例です。

```
# drweb-ctl cfset Update.UpdateInterval 60m
```

- Web インターフェースから実行する場合

  [Settings]メニューから「Updater」を開き、「UpdateInterval」の値を変更します。更新間隔を”60 分”に変更する場合は、”60m”と指定してください。

6.6. フィルターモードでの Lua スクリプト設定

フィルターモードでのメールに対する処理は、"MailD.MilterHook で指定された Lua スクリプトにより実行されます。Lua スクリプトに関しては、下記 URL を参照してください。

https://download.geo.drweb.com/pub/drweb/unix/mail/11.1/documentation/html/en/dw_9_maild_lua.htm

スクリプトは上から順に処理されます。"action ="で指定されたアクション("accept"、"discard"、"reject"、"tempfail"、"replycode")が適用された場合には、以降の条件および処理は適用されませんので、指定順には注意してください。

Postfix とフィルターモードで連携している場合は、「MailD(MailD)」の MilterHook で指定します。

※ SpamdReportHook、RspamdHook は関係ありません。

6.6.1 処理の指定例

- 脅威を含むメールを拒否 ※reject

```
for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do

    return {action = 'reject'}

end
```

または、

```
if ctx.message.has_threat{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} then

    return {action = 'reject'}

end
```

- 脅威を含むメールを圧縮 ※repack

```
for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification", "unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do

    ctx.modifier.repack()

end
```

または、

```
if ctx.message.has_threat{category = {"known_virus", "virus_modification", "unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} then

    ctx.modifier.repack()

end
```

※ どちらの場合も、別途アクションの指定が必要です。

- 記載されている URL が指定したカテゴリに該当する場合、メールを拒否 ※reject

```
for url in ctx.message.urls{category = {"infection_source", "not_recommended", "owners_notice", "adult_content", "violence", "weapons", "gambling", "drugs", "obscene_language", "chats", "terrorism", "free_email", "social_networks", "online_games", "anonymizers", "cryptocurrency_mining_pools"}} do

    return {action = 'reject'}

end
```

または、

```
if ctx.message.has_url{category = {"infection_source", "not_recommended", "owners_notice", "adult_content", "violence", "weapons", "gambling", "drugs", "obscene_language", "chats", "terrorism", "free_email", "social_networks", "online_games", "anonymizers", "cryptocurrency_mining_pools"}} then

    return {action = 'reject'}

end
```

- 記載されている URL が指定したカテゴリに該当する場合、メールを圧縮 ※repack

```
for url in ctx.message.urls{category = {"infection_source", "not_recommended", "owners_notice", "adult_content", "violence", "weapons", "gambling", "drugs", "obscene_language", "chats", "terrorism", "free_email", "social_networks", "online_games", "anonymizers", "cryptocurrency_mining_pools"}} do

   ctx.modifier.repack()

end
```

または、

```
if ctx.message.has_url{category = {"infection_source", "not_recommended", "owners_notice", "adult_content", "violence", "weapons", "gambling", "drugs", "obscene_language", "chats", "terrorism", "free_email", "social_networks", "online_games", "anonymizers", "cryptocurrency_mining_pools"}} then

   ctx.modifier.repack()

end
```

※　どちらの場合も、別途アクションの指定が必要です。

- ログにメールに関する情報を出力する

"/var/log/messages"に出力されます。

1)　スパムスコアを出力する場合

```
dw.notice("Spam score: " .. ctx.message.spam.score)
```

【出力例】

```
Milter client a9fa-0863-0d49: Lua: Spam score: 0
```

2)　スパムタイプを出力する場合

```
dw.notice("Spam type: " .. ctx.message.spam.type)
```

※　"legit"(スパムではないメール)、"spam"(スパムメール)、"virus"(アンチスパムエンジンである Vaderetro によりウイルスが検出されたメール)、"bounce"(バウンスメール)の値が出力されます。

【出力例】

```
Milter client a9fa-0863-0d49: Lua: Spam score: legit
```

3)　検出された脅威名を出力する場合

```
dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
```

【出力例】

```
Milter client a37f-5f32-ca92: Lua: EICAR Test File (NOT a Virus!) found in /
```

4)　URL の該当したカテゴリを出力する場合

```
dw.notice("URL found: " .. url .. "(" .. url.categories[1] .. ")")
```

【出力例】

```
Lua: URL found: <URL>(URL のカテゴリ)
```

- 指定したアドレスにもメールを配信

```
return {action = "accept", added_recipients = {"送信先として追加するアドレス"}}
```

※　"added_recipients"は、アクションとして"accept"を指定した場合のみ利用できます。

- スパム判定(スパムスコアが 100 以上)されたメールを圧縮 ※repack

```
if ctx.message.spam.score >= 100 then

    ctx.modifier.repack()

end
```

※　別途アクションの指定が必要です。

- スパムと判定されたメールの件名に"[SPAM]"を追加 ※change_header_field

```
if ctx.message.spam.score >= 100 then

    local old_value = ctx.message.header.value("Subject") or ""
    local new_value = "[SPAM] " .. old_value
    ctx.modifier.change_header_field("Subject", new_value)

end
```

※　別途アクションの指定が必要です。

- 任意のヘッダーの追加 ※add_header_field

```
ctx.modifier.add_header_field("ヘッダー名", "任意の文字列")
```

例)　MSS11 によってスキャンされたメールに、ヘッダーとして”X-Antivirus: Checked by Dr.Web [MailD version: 11.1”を追加する場合

```
ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
```

例)　"X-Drweb-SpamState"ヘッダーを追加する場合

```
ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
```

例)　"X-DrWeb-SpamScore"ヘッダーを追加する場合

```
ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
```

- 指定した送信元(メールアドレスやドメイン)からのメールを拒否 ※reject

複数のドメインやメールアドレスが対象となるかと思いますので、以下のような内容を含むリストファイルを作成してください。

```
.*@test¥.com
.*@test¥.jp
.*@sample¥.org
```

※ リストファイルには、正規表現で記載してください。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file2 = dw.load_array("リストファイル名")

function milter_hook(ctx)

   if rx.search(file2, ctx.from or '') then

      return {action = 'reject'}

   end

   ※リストに該当しないメールに対する処理を指定してください。

end
```

※ リストファイルを更新した場合、設定を反映させるため drweb-configd を再起動してください。

【その他】

アクションとして”accept”を指定するとホワイトリスト的に使用することができます。使用される場合、条件が適用される順序に注意してください。

➢ 指定した送信元(メールアドレスやドメイン)以外のメールを拒否 ※reject

複数のドメインやメールアドレスが対象となるかと思いますので、リストファイルを作成してください。

※ リストファイルには、正規表現で記載してください。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file1 = dw.load_array("リストファイル名")

function milter_hook(ctx)

   if not rx.search(file1, ctx.from or '') then

      return {action = "reject"}

   end

   ※リストに該当しないメールに対する処理を指定してください。

end
```

※ リストファイルを更新した場合、設定を反映させるため drweb-configd を再起動してください。

➢ 特定の文字列を含む件名のメールや、特定の宛先へのメールを拒否 ※reject

ヘッダーとその値を元にメール対するアクションを指定することができます。複数の条件を設定するケースが多いかと思いますので、以下のような内容を含むリストファイルを作成してください。

```
Subject: .*test.*
Subject: .*テスト.*
To: .*@test¥.com
```

※ リストファイルには、正規表現で記載してください。

※ 上記では、件名(Subject)に"test"または"テスト"が含まれるか、宛先(To)ドメインが"test.com"である事を条件としています。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file2 = dw.load_array("リストファイル名")

function milter_hook(ctx)

    if ctx.message.header.search(file2) then

        return {action = "reject"}

    end

    ※リストに該当しないメールに対する処理を指定してください。

end
```

※　リストファイルを更新した場合、設定を反映させるため drweb-configd を再起動してください。

- 指定した拡張子を持つファイルが添付されたメールを拒否　※reject

添付ファイルの拡張子を元にメール対するアクションを指定することができます。複数の条件を設定するケースが多いかと思いますので、以下のような内容を含むリストファイルを作成してください。

```
.*¥.exe
.*¥.com
```

※　リストファイルには、正規表現で記載してください。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file3 = dw.load_array("リストファイル名")

function milter_hook(ctx)

    if ctx.message.has_part{name_re = file3} then

        return {action = "reject"}

    end

    ※リストに該当しないメールに対する処理を指定してください。

end
```

※　リストファイルを更新した場合、設定を反映させるため drweb-configd を再起動してください。

➢ 本文に指定した文字列を含むメールの場合、件名に[SPAM]を追加 ※change_header_field

メール本文に含まれる文字列に対するアクションを指定することができます。複数の条件を設定するケースが多いかと思いますので、以下のような内容を含むリストファイルを作成してください。

```
This email is a test email.
sample
```

※ リストファイルには、正規表現で記載してください。

※ 全角文字や日本語の文字列を指定した場合、文字コードの違いにより条件に一致しない場合があります。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file3 = dw.load_array("リストファイル名")

function milter_hook(ctx)

   if ctx.message.search(file3) then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)

   end

   ※処理を指定してください。

end
```

※ リストファイルを更新した場合、設定を反映させるため drweb-configd を再起動してください。

6.6.2 Milterhook の設定例

本項で記載している設定例は、Web コンソールから MailD.Milterhook を設定する際の例となります。
"/etc/opt/drweb.com/drweb.ini"を直接編集する場合、「"」は、「""」または「'」と入力してください。

- 設定例 1

スパムに対する処理 ： スパムスコアをログに出力し、そのまま配信。
脅威に対する処理 ： 検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。
ヘッダーに対する処理 ： X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"

function milter_hook(ctx)

    if true then

        ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
        ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
        ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")

        dw.notice("Spam score: " .. ctx.message.spam.score)

    end

    for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
        dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
        ctx.modifier.repack()
    end

    return {action = "accept"}

end
```

➢ 設定例 2

スパムに対する処理　：　スパムスコアをログに出力し、そのまま配信。

脅威に対する処理　：　検出された脅威が指定したカテゴリに該当する場合、ログに出力し、宛先アドレスに加え指定したアドレスに repack して配信。

ヘッダーに対する処理　：　X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"

function milter_hook(ctx)

    if true then

        ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
        ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
        ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")

        dw.notice("Spam score: " .. ctx.message.spam.score)

    end

    for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
        dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
        return {action = "accept", added_recipients = {"送信先として追加するアドレス"}}
    end

    return {action = "accept"}

end
```

➢ 設定例 3

スパムに対する処理　：　スコアが 100 以上の場合には、スパムスコアをログに出力し、件名に[SPAM]を追加し、配信。

脅威に対する処理　：　検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack して配信。

ヘッダーに対する処理　：　X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"

function milter_hook(ctx)

   if ctx.message.spam.score >= 100 then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)
      dw.notice("Spam score: " .. ctx.message.spam.score)

   end

   for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
      ctx.modifier.repack()
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
   end

      ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
      return {action = 'accept'}

end
```

➢ 設定例 4

スパムに対する処理　：　スコアが 100 以上の場合には、スパムスコアをログに出力し、件名に[SPAM]を追加し、配信。

脅威に対する処理　：　検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。

URL に対する処理　：メールに記載されている URL が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。

ヘッダーに対する処理　：　X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"

function milter_hook(ctx)

   if ctx.message.spam.score >= 100 then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)
      dw.notice("Spam score: " .. ctx.message.spam.score)

   end

   for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification", "unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do

      ctx.modifier.repack()
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))

   end

   for url in ctx.message.urls{category = {"infection_source", "not_recommended", "owners_notice", "adult_content", "violence", "weapons", "gambling", "drugs", "obscene_language", "chats", "terrorism", "free_email", "social_networks", "online_games", "anonymizers", "cryptocurrency_mining_pools"}} do
      ctx.modifier.repack()
      dw.notice("URL found: " .. url .. "(" .. url.categories[1] .. ")")
   end

      ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
      return {action = 'accept'}

end
```

➢ 設定例 5

指定したアドレスの処理： 送信元アドレスが"/etc/opt/drweb.com/whitelist"に記載されたアドレスやドメインに該当する場合は、無条件で配信。

※ 以降は、送信元アドレスが、"/etc/opt/drweb.com/whitelist"に該当しない場合に適用。

スパムに対する処理 ： スコアが 100 以上の場合には、スパムスコアをログに出力し、件名に[SPAM]を追加し、配信。

脅威に対する処理 ： 検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。

ヘッダーに対する処理 ： X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file1 = dw.load_array("/etc/opt/drweb.com/whitelist")

function milter_hook(ctx)

   if rx.search(file1, ctx.from or '') then

      return {action = 'accept'}

   end

   if ctx.message.spam.score >= 100 then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)
      dw.notice("Spam score: " .. ctx.message.spam.score)

   end

   for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
      ctx.modifier.repack()
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
   end

      ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
      return {action = 'accept'}

end
```

➢ 設定例 6

指定したアドレスの処理： 送信元アドレスが"/etc/opt/drweb.com/blacklist"に記載されたアドレスやドメインに該当する場合は、無条件で拒否。

※ 以降は、送信元アドレスが、"/etc/opt/drweb.com/blacklist"に該当しない場合に適用。

スパムに対する処理 ： スコアが 100 以上の場合には、スパムスコアをログに出力し、件名に[SPAM]を追加し、配信。

脅威に対する処理 ： 検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。

ヘッダーに対する処理 ： X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file1 = dw.load_array("/etc/opt/drweb.com/blacklist")

function milter_hook(ctx)

   if rx.search(file1, ctx.from or '') then

      return {action = 'reject'}

   end

   if ctx.message.spam.score >= 100 then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)
      dw.notice("Spam score: " .. ctx.message.spam.score)

   end

   for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
      ctx.modifier.repack()
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
   end

      ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
      return {action = 'accept'}

end
```

➢ 設定例 7

指定したアドレスの処理： 送信元アドレスが"/etc/opt/drweb.com/whitelist"に記載されたアドレスやドメインに該当する場合は、ウイルスチェックを実行して配信(脅威検出時は repack)。

※ 以降は、送信元アドレスが、"/etc/opt/drweb.com/whitelist"に該当しない場合に適用。

スパムに対する処理 ： スコアが 100 以上の場合には、スパムスコアをログに出力し、件名に[SPAM]を追加し、配信。

脅威に対する処理 ： 検出された脅威が指定したカテゴリに該当する場合、ログに出力し、repack(元のメールを quarantin.zip に圧縮)して配信。

ヘッダーに対する処理 ： X-DrWeb-SpamScore、X-DrWeb-SpamState、X-AntiVirus を追加。

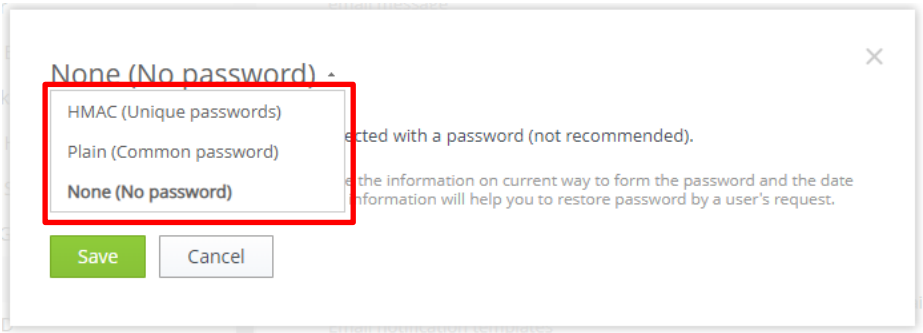
```
local dw = require "drweb"
local rx = require "drweb.regex"
local dwl = require "drweb.lookup"

local file1 = dw.load_array("/etc/opt/drweb.com/whitelist")

function milter_hook(ctx)

   if rx.search(file1, ctx.from or '') then

      for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
         ctx.modifier.repack()
         dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
      end

      return {action = 'accept'}

   end

   if ctx.message.spam.score >= 100 then

      local old_value = ctx.message.header.value("Subject") or ""
      local new_value = "[SPAM] " .. old_value
      ctx.modifier.change_header_field("Subject", new_value)
      dw.notice("Spam score: " .. ctx.message.spam.score)

   end

   for threat, path in ctx.message.threats{category = {"known_virus", "virus_modification",
"unknown_virus", "adware", "dialer", "joke", "riskware", "hacktool"}} do
      ctx.modifier.repack()
      dw.notice(threat.name .. " found in " .. (ctx.message.part_at(path).name or path))
   end

      ctx.modifier.add_header_field("X-AntiVirus", "Checked by Dr.Web [MailD version: 11.1]")
      ctx.modifier.add_header_field("X-DrWeb-SpamScore", ctx.message.spam.score)
      ctx.modifier.add_header_field("X-DrWeb-SpamState", ctx.message.spam.type)
      return {action = 'accept'}

end
```

6.7. “repack“時のパスワードの設定

処理として”repack”を指定した場合、条件に該当するメールは quarantine.zip に圧縮された状態で受信者に配信されます。初期設定では、パスワードは設定されていませんので、パスワードを設定してください。共通のパスワードを使用することも、メール毎にランダムなパスワードを用いることも可能です。

6.7.1　パスワード設定

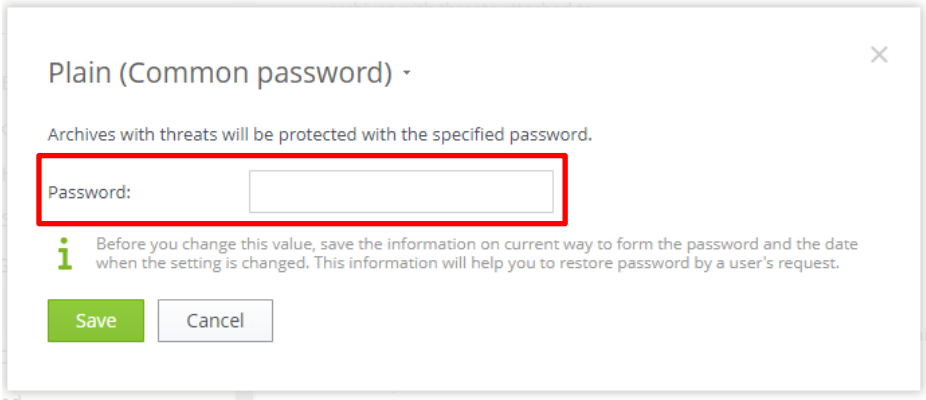
1) Web インターフェースにログインします。
2) 左側のメニューから「Settings」をクリックします。
3) 次に「MailD」をクリックします。
4) 画面右側に表示された「MailD」ページ内の「General」セクションに移動します。
5) 「RepackPassword」の”None”と書かれた箇所をクリックします。
6) 表示された画面で「None (No password)」と書かれている箇所をクリックすると、以下のようなリストが表示されます。



7) 「Plain (Common password)」または「HMAC (Unique password)」を選択します。

- 「Plain (Common password)」を選択した場合

  「Password」欄にパスワードを入力し、「Save」をクリックしてください。

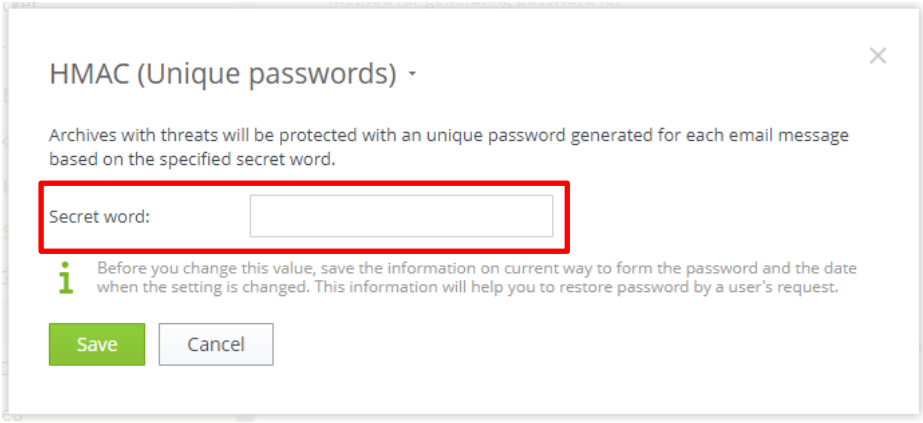


※　”drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset MailD.RepackPassword "Plain(パスワード)"
```

➢ 「HMAC (Unique password)」

「Secret word」欄に任意の文字列を入力し、「Save」をクリックしてください。



※ ”drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset MailD.RepackPassword "HMAC(任意の文字列)"
```

6.7.2 HMAC(ランダムパスワード)指定時のパスワードの取得

「RepackPassword」で HMAC を指定した場合、以下のような本文のメールが受信者に配信されます。

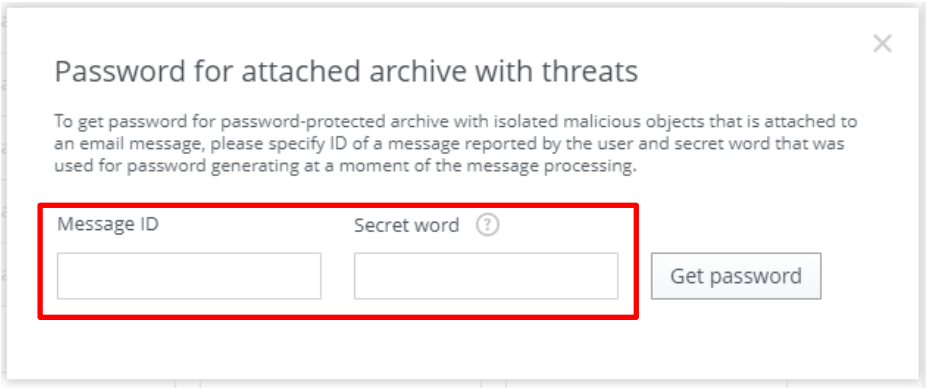
> [Dr.Web Anti-virus] THREAT DETECTED
>
> The original message violated security policies defined by the administrator. It is moved to the password-protected archive attached to this message.
> To get access to the contents of the archive, contact a mail system administrator. You should specify the ID of the received message in your request: 977943.

本文の「the ID of the received message in your request」の後に記載されている数字(コード)が、パスワードを取得する際に必要となります。

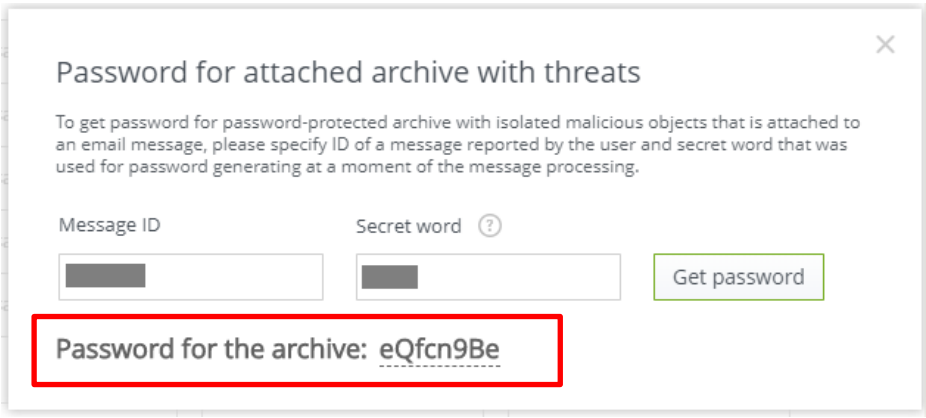
※ 上記の例では、”977943”がコードとなります。

パスワードの取得の手順は、下記となります。

1) Web インターフェースにログインします。
2) 左側のメニューから[Password for attached archive with threats]をクリックします。
3) 表示された「Password for attached archive with threats」画面で、Message ID 欄に配信されたメールに記載されている数字(コード)、Secret word 欄には設定時に指定した任意の文字列を入力します。



4) 「Get Password」をクリックすると、パスワードが表示されます。



※ ”drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl idpass コード
```

6.7.3　管理者メールアドレスの登録

アクションとして REPACK が指定され、添付された圧縮ファイルにパスワードが設定されている場合、以下のようなメールが配信されます。

> [Dr.Web Anti-virus] THREAT DETECTED
>
> The original message violated security policies defined by the administrator. It is moved to the password-protected archive attached to this message.
> To get access to the contents of the archive, contact a mail system administrator. You should specify the ID of the received message in your request: 977943.

※　上記は、HMAC が指定されている場合のものです。

配信されたメール内には、メール管理者(mail system administrator)に連絡するよう記載されていますが、メール管理者のアドレスは記載されておりません。”TemplateContacts”を設定することにより、以下のように、本文中に管理者のメールアドレス(青字の箇所)を表示させることができます。

> [Dr.Web Anti-virus] THREAT DETECTED
>
> The original message violated security policies defined by the administrator. It is moved to the password-protected archive attached to this message.
> To get access to the contents of the archive, contact a mail system administrator: **管理者のメールアドレス**. You should specify the ID of the received message in your request: 335590.

”TemplateContacts”の設定手順は、下記となります。

1) Web インターフェースにログインします。
2) 左側のメニューから「Settings」をクリックします。
3) 次に「MailD」をクリックします。
4) 画面右側に表示された「MailD」ページ内の「Notification Templates」セクションに移動します。
5) 「TemplateContacts」の"Not specified"と書かれた箇所をクリックします。
6) 表示された画面で管理者のメールアドレスを入力し、「Save」をクリックします。

※　”drweb-ctl”コマンドで設定する場合は、下記となります。

```
# drweb-ctl cfset MailD.TemplateContacts 管理者のメールアドレス
```

6.8. ESS サーバとの接続

構築済みの ESS11 サーバに MSS を接続します。ESS サーバがインターネットに接続されていれば、MSS をインストールしたサーバがインターネットに接続していない状態でも、定義ファイルの更新が可能になります。

尚、設定は集中管理サーバー上のものが優先されます。

集中管理サーバの管理画面(ControlCenter)上の操作が必要ですので、アクセスできる状態で実施してください。

6.8.1　コマンドラインから実行する場合

1) ESS11 サーバより drwcsd-certificate.pem(証明書)ファイルをダウンロードします。
   https://<IP アドレス>:9081/install/drwcsd-certificate.pem
   ※ drwcsd-certificate.pem ファイルの内容は、ESS サーバ毎に異なりますので、接続先サーバより入手してください。
2) 以下のコマンドを実行し、ESS11 サーバに接続します。

```
# drweb-ctl esconnect --Certificate <path>drwcsd-certificate.pem <ESS サーバアドレス>:2193
```

例) ESS11 サーバのアドレスが 192.168.1.126、drwcsd.pub を/home/test に保存している場合

```
# drweb-ctl esconnect --Certificate /home/test/drwcsd-certificate.pem 192.168.1.126:2193
```

※ 接続先サーバの IP アドレスやポートを誤って指定した場合は、以下のコマンドを実行し切断した後に再度実施してください。

```
# drweb-ctl esdisconnect
```

3) ESS11 サーバに接続されると「Pending ・・・」のメッセージが表示され、承認されると「Accepted by ・・・」のメッセージが表示されます。

```
Pending for approval from central protection server
Accepted by tcp:// <ESS サーバアドレス>:2193
```

4) ブラウザから ControlCenter にログインします。
5) 「アンチウィルスネットワーク」メニュー中央のツリーから、[Status]-[Newbies]を開きます。
6) 表示されている端末(MSS をインストールしたサーバ名が表示されます)を選択し、承認します。
7) 「アンチウィルスネットワーク」メニュー中央のツリーから、[Everyone]を開き、MSS をインストールしたサーバのアイコンが緑色の状態であることを確認します。
8) MSS をインストールしたサーバ上の/var/opt/drweb.com/bases/drwtoday.vdb が、更新されていることを確認します。
   ※ ESS11 サーバと切断する場合(集中管理から外す場合)は、以下のコマンドを実行してください。

```
# drweb-ctl esdisconnect
```

6.8.2　Web インターフェースから実行する場合

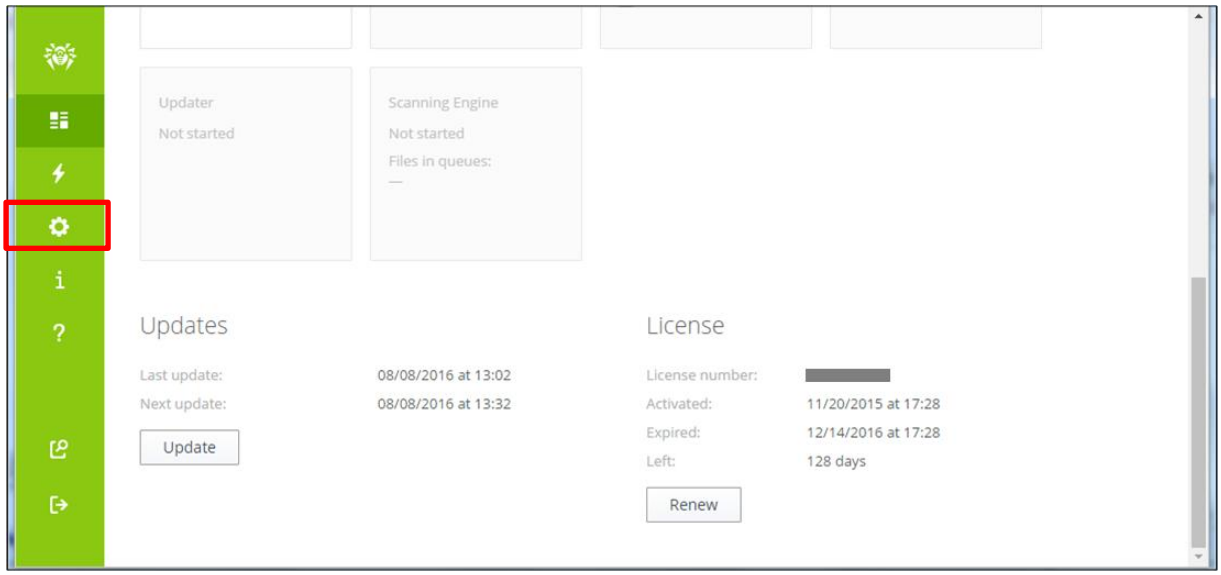
1)　ESS11 サーバより drwcsd.pub(公開鍵)ファイルをダウンロードします。

https://<IP アドレス>:9081/install/drwcsd.pub

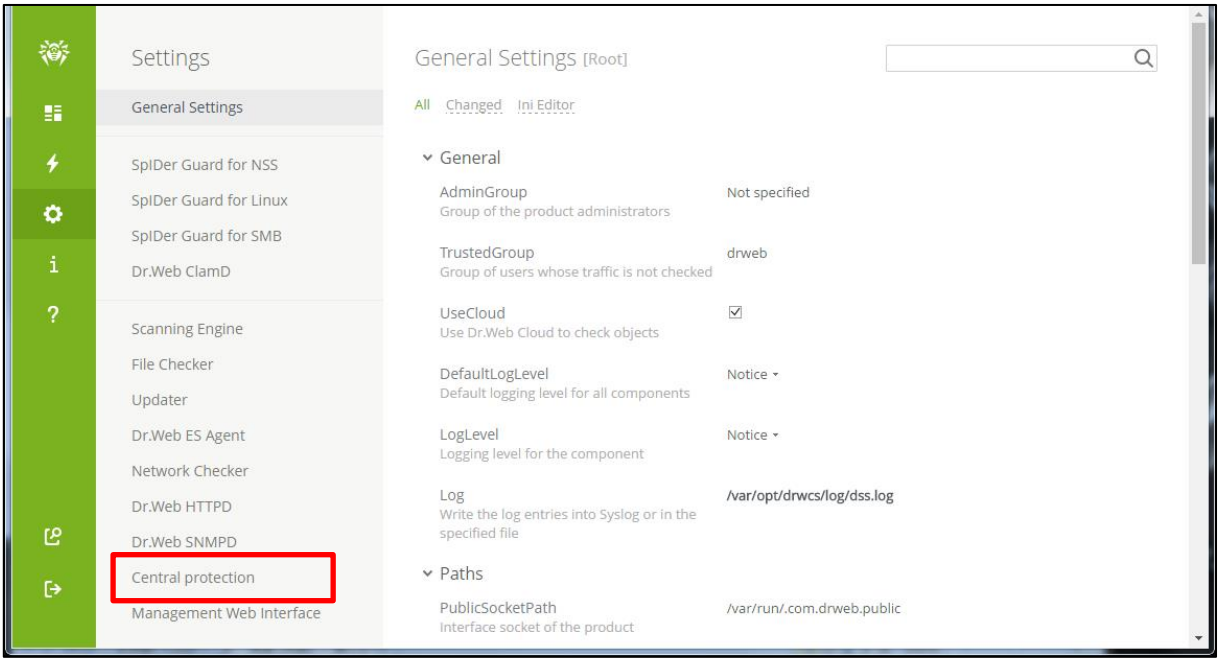
※　drwcsd.pub ファイルは ESS サーバ毎に異なりますので、接続先サーバより入手してください。

2)　Web インターフェースにログインします。

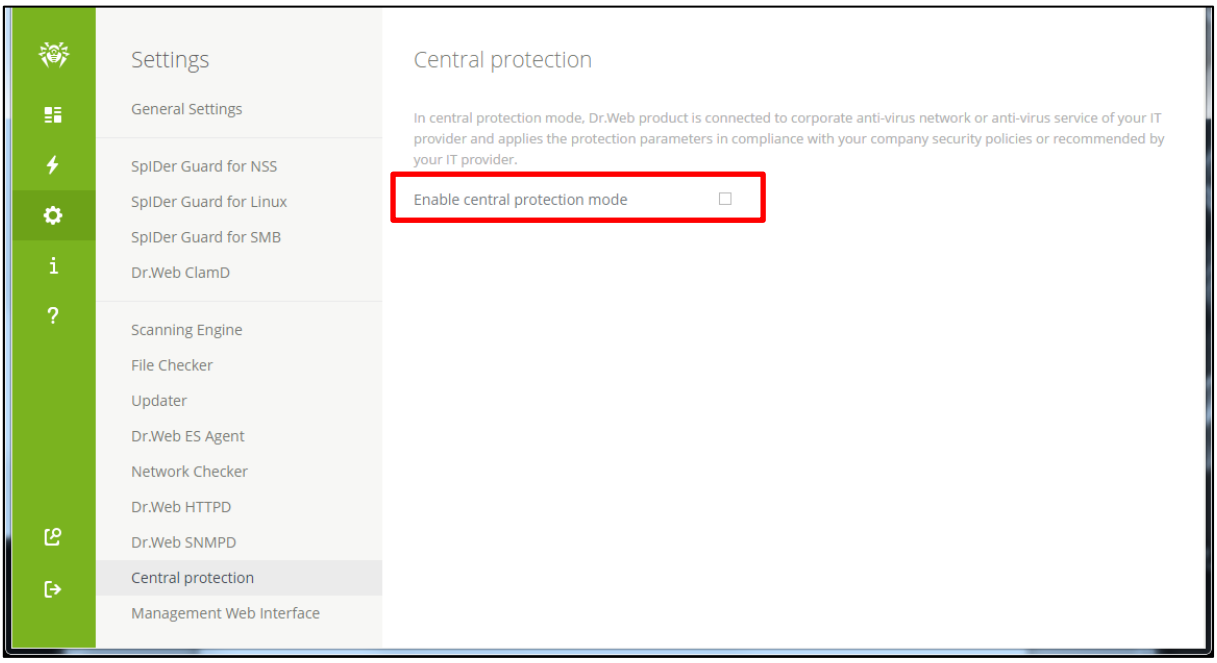
3)　[Settings]をクリックします。



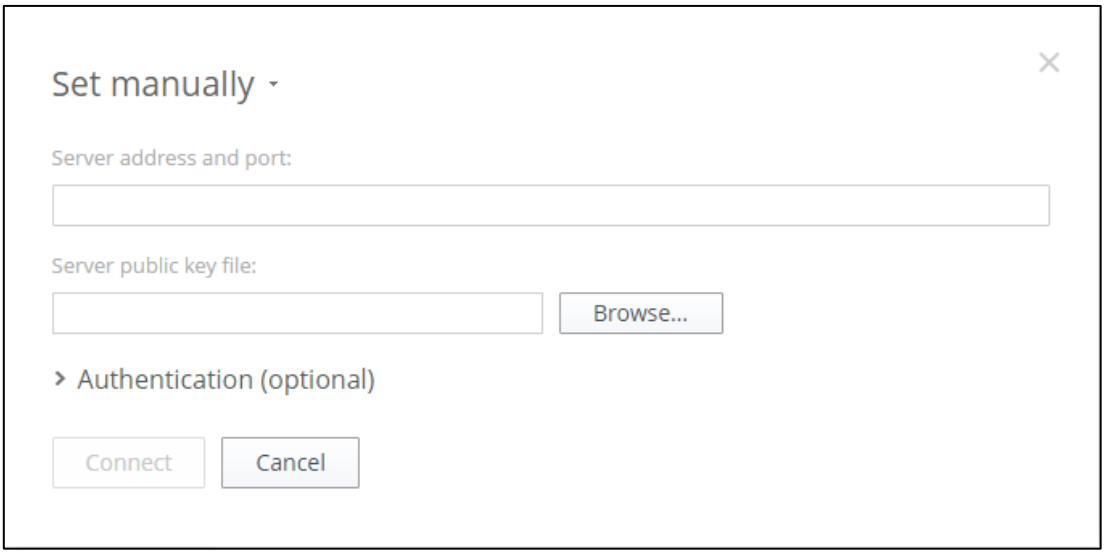
4)　[Central protection]をクリックします。



5) “Enable central protection mode”にチェックを入れます。



6) 接続先サーバとポート番号(IP アドレス:2193)を指定し、「Browse」ボタンをクリックして drwcsd.pub を指定した後、「Connect」ボタンをクリックします。

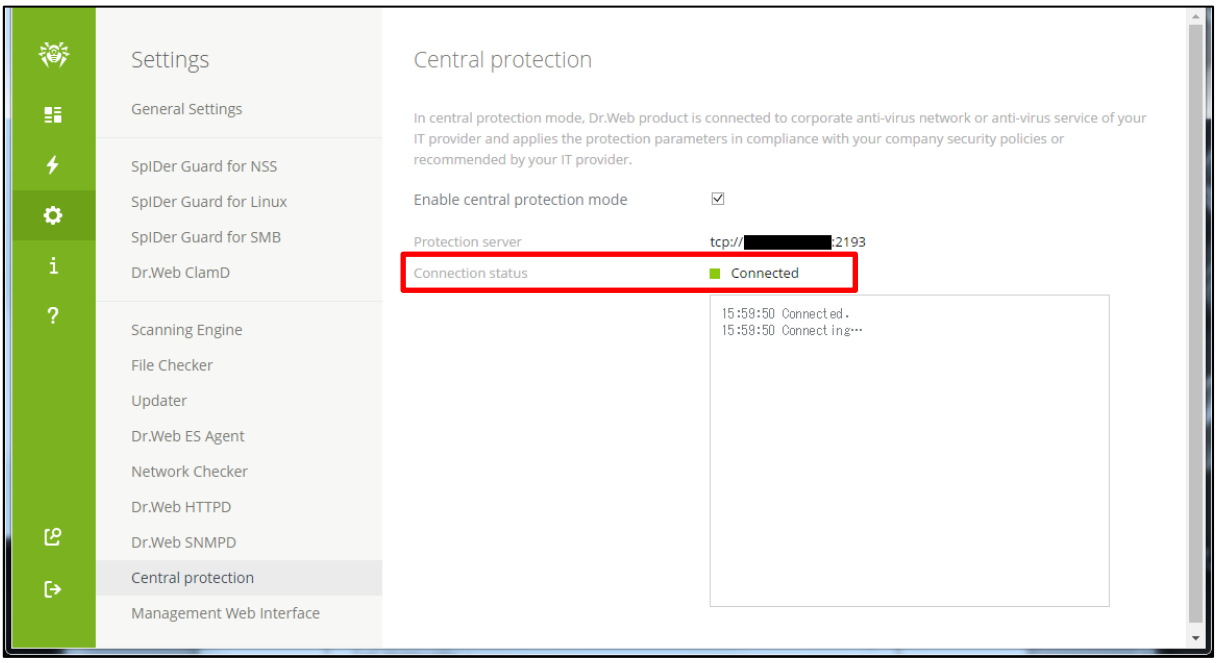


7) ブラウザから ControlCenter にログインします。

8) 「アンチウィルスネットワーク」メニュー中央のツリーから、[Status]-[Newbies]を開きます。

9) 表示されている端末(MSS をインストールしたサーバ名が表示されます)を選択し、承認します。

10) Web インターフェース上で「Connection status」が「Connected」と表示されていることを確認します。



※ ESS11 サーバと切断する場合(集中管理から外す場合)は、"Enable central protection mode"のチェックを外してください。

6.9. 以前のバージョンの MSS のアンインストール

以下は、MTA として postfix を使用し、drweb-mail-servers を利用している場合のアンインストール手順となります。

1) postfix のプロセスを停止します。

2) 以下のコマンドを実行します。

```
# /opt/drweb/remove.sh
```

3) 以下のメッセージが表示されたら、「YES」と入力して、「Enter」キーを押します。

```
This script will help you remove Dr.Web packages


Do you want to continue? (YES/no)
```

4) 以下のメッセージが表示されたら、「A」と入力して、「Enter」キーを押します。

```
Select the software you want to remove:
        [ ] 1 Dr.Web Agent - Additional files to run Dr.Web Agent in central protection mode (6.0.2.4)
        [ ] 2 Dr.Web Agent (6.0.2.4)
~~ 略 ~~
        [ ] 22 Dr.Web Maild Web Interface (6.0.2.2)
        [ ] 23 Dr.Web Mail Daemon (6.0.2.8)
        [ ] 24 Dr.Web Monitor (6.0.2.3)
        [ ] 25 Dr.Web Antivirus Scanner (6.0.2.3)
        [ ] 26 Dr.Web Updater (6.0.2.7)

To select a package you want to remove or deselect some previously
selected package - enter the corresponding package number and press Enter.

You may enter A or All to select all the packages, and N or None to deselect all of them.
Enter R or Remove to remove selected packages.
Enter 0, Q or Quit to quit the dialog.
All values are case insensitive.
Select:
```

5) 全ての項目が「X」となっていることを確認し、「R」と入力して、「Enter」キーを押します。

```
Select the software you want to remove:
        [X] 1 Dr.Web Agent - Additional files to run Dr.Web Agent in central protection mode (6.0.2.4)
        [X] 2 Dr.Web Agent (6.0.2.4)
~~ 略 ~~
        [X] 22 Dr.Web Maild Web Interface (6.0.2.2)
        [X] 23 Dr.Web Mail Daemon (6.0.2.8)
        [X] 24 Dr.Web Monitor (6.0.2.3)
        [X] 25 Dr.Web Antivirus Scanner (6.0.2.3)
        [X] 26 Dr.Web Updater (6.0.2.7)

To select a package you want to remove or deselect some previously
selected package - enter the corresponding package number and press Enter.

You may enter A or All to select all the packages, and N or None to deselect all of them.
Enter R or Remove to remove selected packages.
Enter 0, Q or Quit to quit the dialog.
All values are case insensitive.
Select:
```

6) 以下のメッセージが表示されたら、「YES」と入力して、「Enter」キーを押します。

```
A list of packages marked for removal:
        drweb-agent-es
        drweb-agent
~~ 略 ~~
        drweb-maild
        drweb-monitor
        drweb-scanner
        drweb-updater
Are you sure you want to remove the selected packages? (YES/no)
```

7) 以下のメッセージが表示されたことを確認します。

```
Removing empty installation directories...
Removal of drweb-updater is complete.
#
```

8) "/etc/postfix/main.cf"に追加されている、Dr.Web との連携用の行をコメントアウトします。

```
########################################
### ADDED BY MAILD-POSTFIX INSTALL ###
########################################
content_filter = scan:127.0.0.1:8025
receive_override_options = no_address_mappings
```

9) "/etc/postfix/master.cf"に追加されている、Dr.Web との連携用の以下の行をコメントアウトします。

```
########################################
### ADDED BY MAILD-POSTFIX INSTALL ###
########################################
scan        unix  -       -       n       -       -       smtp
            -o smtp_send_xforward_command=yes
127.0.0.1:8026 inet  n       -       n       -       -       smtpd
            -o content_filter=
            -o receive_override_options=no_unknown_recipient_checks,no_header_body_checks
            -o smtpd_helo_restrictions=
            -o smtpd_client_restrictions=
            -o smtpd_sender_restrictions=
            -o smtpd_recipient_restrictions=permit_mynetworks,reject
            -o mynetworks=127.0.0.0/8
            -o smtpd_authorized_xforward_hosts=127.0.0.0/8
```

10) postfix のプロセスを起動し、メールの送受信ができることを確認します。

お使いの製品の詳細な機能の説明や、利用方法は、各製品マニュアルをご参照ください。
また、製品のご利用について、ご質問やトラブル等がありましたら、下記 URL よりお気軽にお問い合わせください。

https://support.drweb.co.jp/support_wizard/

株式会社 Doctor Web Pacific
〒105-0003 東京都港区西新橋 1-14-10 西新橋スタービル 2F
URL：www.drweb.co.jp