

sudoを使用したスーパーユーザ特権の設定

概要

sudo設定の基本を理解し、**sudo**を使用してスーパーユーザ特権を委任する方法を習得します。

目的

コマンドによっては、管理者特権またはroot特権が必要なものがあります。**sudo**を使用すると、そのようなコマンドを実行する特権を特定のユーザやグループに委任できます。

所要時間

この記事の理解には20分ほどを要します。最初の**sudo**設定ルールの作成に要する時間は数分たらずですが、環境全体で機能する**sudo**設定を確立するには、設定の複雑さに応じて相応の時間を要します。

目標

sudo設定の基本的な特性を理解します。**sudo**設定の一般的な使用例に対処します。**sudo**セットアップでユーザ、ユーザグループ、およびエイリアスを操作する方法を習得します。**sudo**のベストプラクティスとトラブルシューティングを理解します。

要件

- sudoの基本的な理解。
- root特権。通常のユーザとしてsudoを使用する方法については<https://documentation.suse.com/smart/systems-management/html/sudo-run-commands-as-superuser/index.html>を参照してください。
- sudoパッケージをインストールする必要があります。

発行日: 11/12/2025

目次

- 1 **sudo**の設定の概要 3
- 2 カスタム**sudo**設定の作成 4
- 3 **sudo**のパスワードプロンプトのタイムアウト変更 9
- 4 root特権によるシェルの起動 10
- 5 **sudo**のベストプラクティス 11
- 6 トラブルシューティング 12
- 7 **sudo**設定のリファレンス 13
- 8 法的事項 17
- A GNUフリー文書利用許諾契約書(GFDL) 17

1 sudoの設定の概要

sudoは、特定のユーザまたはグループにスーパーユーザ特権を安全かつ効率的に委任する手段を提供します。

Linuxシステムでは、特定の操作にroot特権または管理者特権が必要です。ホームユーザが自身のシステムを管理する場合、スーパーユーザ特権を委任する必要はありません。このシナリオでは、管理者と通常のユーザが同一人物であるためです。ただし、システムが複数のユーザ、グループ、およびホストで構成される大規模なシステム環境の一部になるとすぐに、誰がどこで何を実行できるかを常に制御することが不可欠になります。同時に、すべてのユーザとグループにそのタスクを実行するために必要な特権を付与することも重要になります。

sudoは、このような環境を支援するように設計されています。次のような機能が得られます。

システムセキュリティの強化

sudoを使用すると、ユーザ、グループ、ホスト、およびコマンドをきめ細かく制御できるため、侵入者やシステムユーザによる悪意のある損害や偶発的な損傷を受けるリスクが減り、システムのセキュリティが向上します。

優れた監査証跡

これは、ユーザが特権を切り替えるたびにシステムのログに表示され、このユーザが昇格された特権で実行したすべての操作を遡って追跡できます。

root固有のタスクを委任する方法

sudoを使用すると、システム管理者は、単一のユーザやグループがrootパスワードを入力したり、rootアカウントに切り替えたりすることなく、特定のタスクを実行できるようにすることができます。

！ 重要: この記事の利用方法

この記事では、**sudo**の詳しい設定情報を提供します。ただし、包括的で安全な**sudo**ポリシーの構築方法に関するアドバイスは提供しません。セキュリティ関連のポリシーは非常に複雑であり、ポリシーの作成対象の環境によって大きく異なります。

2 カスタム**sudo**設定の作成

簡単なカスタムの**sudo**設定例を構築し、それを段階的に拡張する方法について習得します。グループを作成し、エイリアスを使用することで、カスタム設定の簡潔性と効率性を維持します。

🚫 警告: 設定例はデモンストレーションのみを目的としています

以下で紹介するルール例はデモンストレーションのみを目的としています。これらのルール例を利用して、**sudo**設定ファイルの一般的な構文を理解してください。実際の環境では、このルール例をそのまま使用しないでください。環境の複雑さを反映していないからです。

2.1 **sudo**の設定でのベストプラクティス

まず、**sudo**設定を維持するための基本ルールについて説明します。

sudoの設定ファイルの編集には必ず**visudo**を使用する

sudoの設定の変更では、どの場合も**visudo**コマンドを使用する必要があります。**visudo**は、**sudo**設定ファイルを編集できる特別なツールです。また、基本的な構文チェックを実行して、設定が壊れておらず機能することを確認することもできます。**sudo**の設定に不備があると、ユーザが自身のシステムからロックアウトされることがあります。

必ず`/etc/sudoers.d/`にカスタム設定を作成する

カスタム設定は、**sudo**によって取得できるように、`/etc/sudoers.d/`に置く必要があります。カスタム設定ファイルに記述した設定は、`/etc/sudoers`のデフォルト設定よりも優先されます。

設定が読み取られる順序にいつでも注意する

カスタム設定が確実に正しい順序で読み取られるように数字のプレフィックスを付加します。先頭に0を付加してファイルの読み取り順序を指定します。これにより、たとえば01_myfirstconfigは10_myotherconfigよりも前に解析されます。順番に読み取られる複数のディレクティブを設定したファイルで、これらの各ディレクティブに記述された情報が互いに矛盾していると、最後に読み込まれたディレクティブが適用されます。

必ずわかりやすいファイル名を使用する

設定ファイルの機能がわかるようなファイル名を使用します。`sudo`のセットアップで想定している動作を追跡する際に、この措置が効果的です。



ヒント: `sudo`設定と不変ファイルシステム

不変ファイルシステムとは、一度インストールしたら変更できないファイルシステムのことです。このファイルシステムは読み込み専用でアクセスされます。使用しているSUSE製品が不変ファイルシステムに依存する場合、製品に付属するデフォルトの`sudo`設定は`/usr/etc/sudoers`にインストールされ、事前設定の調整は`/usr/etc/sudoers.d/`にあります。

独自のカスタム設定は`/etc/sudoers.d/`に置かれ、`/usr/etc/sudoers.d/`で指定された設定よりも優先されます。`visudo`コマンドは、`/usr/etc/sudoers`を開き、古い`sudoers`ファイルがなければ、変更されたファイルを`/etc/sudoers`に保存します。ファイルがすでに存在していた場合、`visudo`はそのファイルを開き、そのファイルに書き込みます。`/etc/`にあるインスタンスは、`/usr/etc/`に保存されているインスタンスよりも優先されます。こうすることで、ユーザが行った設定の調整が更新時に壊れないようにします。

2.2 ユーザ固有の設定ファイルの作成

通常ユーザ(tux)が、`root`パスワードではなく自身のパスワードを使用して、`useradd`コマンドを使用できるように、`sudo`の設定ファイルを作成します。

例 1: ユーザ固有の設定ファイルの作成

1. 新しいユーザ固有のディレクティブを保持するカスタム設定ファイルを作成します。そのためには、システム管理者(`root`)として`visudo`を起動します。番号とわかりやすい名前の両方を使用します。

```
# visudo -f /etc/sudoers.d/02_usermanagement
```

2. この`sudo`設定を適用する環境全体で`tux`が`/usr/sbin/useradd`バイナリを実行できるようにするルールを作成します。

```
tux ① ALL ② = /usr/sbin/useradd ③
```

- ① ユーザまたはグループを指定します。ユーザは名前または#UIDで一覧にし、グループは%GROUPNAMEで一覧にします。複数の項目はコンマで区切って記述します。エントリを無効にするには!を使用します。
- ② ホストを1つまたはコンマで区切って複数指定します。完全修飾ホスト名またはIPアドレスを使用します。すべてのホストにこの設定をグローバルに適用するにはALLを追記します。適用しない場合は!を使用します。
- ③ 実行可能ファイルを1つまたはコンマで区切って複数指定します。各ファイルの指定では次のルールに留意します。

`/usr/sbin/useradd`

追加オプションを追記しない場合は、実行可能なすべての`useradd`コマンドを実行できます。

`/usr/sbin/useradd -c`

明示的にオプションを指定すると、そのオプションのみが適用されます。上記で指定したユーザは、これ以外のオプションを利用できません。

`/usr/sbin/useradd ""`

オプションを指定せずに`useradd`の呼び出しのみができるようにします。

上記の例では、すべてのオプションとサブコマンドを許可するか、またはセキュリティ上の理由から数個に制限したいと考えていますが、ユーザによるオプションの指定を一切禁止することは、この状況では無意味です。

3. ユーザが`root`パスワードではなく自身のパスワードを使用できるようにするには、次の行を追加します。

```
Defaults:tux !targetpw
```

このフラグがアクティブな場合、目的のユーザ(`root`)のパスワードの入力が求められます。SLE Microシステムでは、このフラグがデフォルトで有効になっています。`!`を使用してこれを否定し、ユーザに`root`パスワードではなく自身のパスワードの入力を要求します。

4. 設定を保存してエディタを終了し、2つ目のシェルを開いて、`sudo`が新しい設定に従っているかどうかをテストします。

2.3 項目のグループ化によるカスタム設定の作成

指定したユーザのグループがrootパスワードを必要とせずにuseraddコマンドを実行できるように、例1「ユーザ固有の設定ファイルの作成」の設定を変更します。また、このグループで利用可能なコマンドのリストにusermodとuserdelを追加します。

例 2: 項目のグループ化によるカスタム設定の作成

1. この設定例を変更するには、visudoを使用してシステム管理者として設定を開きます。

```
# visudo /etc/sudoers.d/02_usermanagement
```

2. コンマ区切りで記述した複数のユーザをルールに追加します。

```
tux, wilber ALL = /usr/sbin/useradd
```

3. ここで記述したユーザが複数のコマンドを実行できるようにするには、それらのコマンドをコンマ区切りで指定します。

```
tux, wilber ALL = /usr/sbin/useradd, /usr/sbin/usermod, /usr/sbin/userdel
```

4. 一覧にしたユーザがrootパスワードではなく自身のパスワードを使用できるようにするには、次の行を追加します。

```
Defaults:tux, wilber !targetpw
```

このフラグがアクティブな場合、目的のユーザ(root)のパスワードの入力が求められません。SLE Microシステムでは、このフラグがデフォルトで有効になっています。!を使用してこれを否定し、一覧にしたユーザにrootパスワードではなく自身のパスワードの入力を要求します。

5. 設定を保存してエディタを終了し、2つ目のシェルを開いて、sudoが新しい設定に従っているかどうかをテストします。

2.4 エイリアスの適用による設定の簡潔化

エイリアスを使用して、例2「項目のグループ化によるカスタム設定の作成」のカスタム設定の簡素化を進めます。項目をグループ化することにはある程度は役に立ちますが、ユーザ、コマンド、およびホストにグローバルエイリアスを使用すると、クリーンで簡潔なsudo設定を最も効率的に維持できます。

リストではなくエイリアスとグループを使用する方が、より効果的にセットアップの変更に対処できます。グループからユーザが離れる場合は、独立したカスタム設定ファイルをすべて調べるのではなく、`User_Alias`のグローバル宣言からそのユーザを削除するだけで済みます。他のタイプのエイリアス(`Host_Alias`、`Cmnd_Alias`、および`Runas_Alias`)にも同じ手順が当てはまります。

例 3: エイリアスの適用による設定の簡潔化

1. グローバルエイリアス定義を保持する新しいファイルを作成します。

```
# visudo /etc/sudoers.d/01_aliases
```

2. 次の行を追加して、`TEAMLEADERS`エイリアスを作成します。

```
User_Alias    TEAMLEADERS = tux, wilber
```

3. 次の行を追加して、`USERMANAGEMENT`エイリアスを作成します。

```
Cmnd_Alias    USERMANAGEMENT = /usr/sbin/useradd, /usr/sbin/usermod, /usr/sbin/userdel
```

4. 変更を保存して、`visudo`を終了します。

5. システム管理者として`visudo`を起動して、設定ファイル例を次のように編集します。

```
# visudo -f /etc/sudoers.d/02_usermanagement
```

6. 以前のルールを削除し、上記で定義したエイリアスを使用する次のルールに置き換えます。

```
TEAMLEADERS ALL = USERMANAGEMENT
```

7. `User_Alias`で定義されているすべてのユーザが`root`パスワードではなく自身のパスワードを使用できるようにするには、次の行を追加します。

```
Defaults:TEAMLEADERS !targetpw
```

8. 設定を保存してエディタを終了し、2つ目のシェルを開いて、`sudo`が新しい設定に従っているかどうかをテストします。



注記: 詳細情報

7項「[sudo設定のリファレンス](#)」で`sudo`設定構文の詳細な説明を参照するか、`sudo`のマニュアルページを参照してください。

3 sudoのパスワードプロンプトのタイムアウト変更

コマンドごとにrootパスワードの入力を求められることなく、root特権を必要とするコマンドを実行する際のタイムアウト設定を変更する方法について説明します。

先頭がsudoであるコマンドを初めて実行するときにrootパスワードの入力を求められます。このパスワードは一定期間有効です。この期間が経過すると、パスワードの入力を再度求められます。root特権を必要とするコマンドを実行する際のこのタイムアウトを延長または短縮するには、sudo設定ファイルを次のように変更します。



警告: root特権にパスワードを使用しないアクセスを無制限に許可しないようにします

セキュリティ上の理由から、root特権に無制限のアクセスを許可しないでください。代わりに、侵入者によるrootアカウントの悪用を防止するために適切なタイムアウトを設定します。

手順 1: sudoのパスワードプロンプトのタイムアウト変更

1. システム管理者として、次のようにtimestamp設定を指定した新しいsudo設定ファイルを作成します。

```
# visudo --f=/etc/sudoers.d/timestamp_timeout
```

rootパスワードによって正しく認証されると、この設定ファイルが開きます。

2. 編集を有効にし、行timestamp_timeout=を追加します。timestampの値を入力します。たとえば、タイムアウトを3分に短縮するには次のように指定します。

```
timestamp_timeout=3
```

タイムアウトを0に設定していると、sudoコマンドを実行するたびにrootパスワードの入力を求められます。

3. 変更を保存して、ファイルを閉じます。

sudo設定ファイルを作成し、sudoコマンドの実行のタイムアウト設定を短くしました。

4 root特権によるシェルの起動

`sudo -s`コマンドまたは`sudo -i`コマンドを使用して、永続的なroot特権でシェルを起動します。どちらのコマンドでも、rootパスワードの入力を一度だけ求められます。

4.1 `sudo -s`と`sudo -i`の違い

rootとしてコマンドを実行するたびに`sudo`の入力が必要であると煩雑な作業になることがあります。代わりに、いずれかの組み込み機能を使用して永続的なroot特権でシェルを起動できます。この方法では次の2つのコマンドオプションを利用できます。

- `sudo -s`では、現在のユーザの環境でシェルを起動し、いくつかの特権制御手段も使用できます。このコマンドを実行するにはrootパスワードを入力します。
- `sudo -i`: クリーンな環境で対話型ログインシェルとしてシェルを起動します。このコマンドを実行するにはrootパスワードを入力します。

どちらのコマンドでも、シェルは新しい環境で起動し、ユーザはrootとしてログインします。起動したシェルで実行する各コマンドは、パスワードの再入力を必要としない昇格した特権で実行されます。この環境はシェルを閉じると終了するため、別の`sudo`コマンドでは再度パスワードを入力する必要があります。

4.2 `sudo -s`によるシェルの起動

`sudo -s`コマンドでは対話型の非ログインシェルが起動します。rootパスワードによって正しく認証されると、以降のすべてのコマンドは昇格した特権で実行されます。

SHELL環境変数またはユーザのデフォルトのシェルで、どのシェルを開くかを指定します。この変数が空の場合は、`/etc/passwd`で定義したシェルが選択されます。

デフォルトでは、現在のユーザが前のユーザの環境を継承することから、前のユーザのディレクトリから`sudo -s`コマンドが実行されます。このコマンドは履歴にも記録されます。

永続的に昇格した特権でシェルを起動するには次のコマンドを入力します。

```
tux:~ > sudo -s
[sudo] password for root:
root:/home/tux # exit
```

```
tux:~ >
```

プロンプトが`>` から`#` に変わります。

これで、永続的に昇格した特権でシェルが起動しています。以降のすべてのコマンドは、パスワードの再入力を求められることなく実行できます。

4.3 `sudo -i`によるシェルの起動

`sudo -i`は`sudo -s`コマンドラインオプションに似ていますが、起動するシェルは対話型ログインシェルです。`sudo -s`コマンドを使用すると、現在のユーザは前のユーザの環境を継承します。`sudo -i`コマンドを使用すると、これを防ぐことができます。この場合、ターゲットユーザはクリーンな環境を取得し、専用の`$HOME`ディレクトリで起動します。

`sudo -i`でコマンドを実行するには次のように入力します。

```
tux:~ > sudo -i
[sudo] password for root:
root:~ # exit
tux:~ >
```

永続的に昇格された特権でシェルが起動し、コマンドが履歴に記録されます。以降のすべてのコマンドは、パスワードの再入力を求められることなく実行できます。

5 `sudo`のベストプラクティス

システムアクセスを制御し、ユーザの生産性を向上させるための`sudo`のベストプラクティスをいくつか説明します。

`sudo`設定を徹底的にテストおよび監査します。

真に効率的で安全な`sudo`設定フレームワークを構築するには、定期的なテストと監査を定型業務として確立します。考えられる抜け穴を特定し、それに対処します。セキュリティよりも利便性を優先しないようにします。

独立したファイルにカスタム`sudo`設定を保存する

`sudo`向けの主なポリシー設定ファイルは、`/etc/sudoers`です。このファイルはシステムパッケージで提供されており、変更すると更新が動作しなくなる可能性があります。したがって、カスタム設定を記述した別の設定ファイルを`/etc/sudoers.d/`ディレクトリに作成します。`/etc/sudoers`のディレクティブによって、これらのファイルがデフォルトで取り込まれます。

sudoのタイムアウトを制限する

セキュリティ上の理由から、`root`特権に無制限のアクセスを許可しないでください。代わりに、侵入者による`root`アカウントの悪用を防止するために適切なタイムアウトを設定します。詳細については、3項「[sudoのパスワードプロンプトのタイムアウト変更](#)」を参照してください。

visudoコマンドを使用する

`/etc/sudoers`ファイルを安全に編集するには`visudo`コマンドを使用します。変更を保存する前に、このコマンドによってファイルの構文が確認されるからです。これは、システム中断の原因となりがねないエラーを修正するための予防的な手段です。基本的な構文チェックのほかに、`visudo -c`を実行して、`sudo`設定のフレームワーク全体が正しい順序で解析されていてエラーがないかどうかをチェックすることもできます。

個人ではなくグループでユーザを管理する

`sudo`設定をできるだけ簡潔で管理しやすい状態に保ちます。ユーザをグループに追加し、個人ではなくそのグループに特権を付与することによって、ユーザを管理します。これにより、設定全体でユーザを探すことなく、グループ設定を変更するだけでユーザの追加や削除ができます。

`%admingrp`グループに属するすべてのユーザがすべてのコマンドを実行できるようにするルールの例を以下に示します。

```
%admingrp ALL = (ALL) ALL
```

バイナリを実行できるパスを制限する

`secure_path`ディレクティブを使用して、ユーザがコマンドを実行できる領域を制限します。次の例は、SLE Microにデフォルトで用意されている設定です。

```
Defaults secure_path="/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/bin:/usr/local/sbin"
```

sudoログ記録の透過性を維持する

`sudo`では、ログのエントリを見失いやすい標準ログファイルにログが記録されます。設定に次のルールを追加して専用の`sudo`ログファイルを指定します。

```
Defaults logfile=/var/log/sudo.log
```

6 トラブルシューティング

`sudo`設定の問題をデバッグおよびトラブルシューティングする方法について説明します。

6.1 /etc/sudoers.d/にあるカスタム設定の無視

`/etc/sudoers`の`#includedir`ディレクティブでは、ファイル名の末尾が`~`文字であるファイルとファイル名に`.`文字が使用されているファイルが無視されます。これは、パッケージマネージャに用意されている設定ファイル(ファイル名に`.`が使用されているファイル)と、エディタの一時ファイルやバックアップファイル(ファイル名の末尾が`~`であるファイル)に伴う問題を回避するための措置です。カスタム設定ファイルのファイル名にこのような文字を使用していないこと、ファイル名がこのような文字で終わっていないことを確認します。このような文字を使用しているファイル名、末尾がこのような文字であるファイル名は変更します。

6.2 カスタムディレクティブの競合

設定ファイルを読み取る順序によって、`sudo`設定ディレクティブが適用されるタイミングが決まります。`/etc/sudoers.d/`にあるファイルに記述したディレクティブは、`/etc/sudoers`にある同じディレクティブより優先されます。`/etc/sudoers.d/`に記述したカスタムディレクティブが機能しない場合は、`visudo -c`を使用してファイルの読み取り順序を確認します。必要に応じて、この順序を調整します。

6.3 破損したsudo設定によるロックアウト

誤ってシステムの`sudo`設定を破損してしまい、`sudo`からロックアウトされた場合は、`su -`とrootパスワードを使用してルートシェルを起動します。`visudo -c`を実行してエラーがないかどうかをチェックし、`visudo`を使用してエラーを修正します。

7 sudo設定のリファレンス

このセクションでは、デフォルトおよびカスタムの両方の`sudo`設定を理解および管理するのに役立つ基本的な`sudo`設定リファレンスを提供します。

7.1 sudoers設定の構文

`sudoers`の設定ファイルには、2種類のオプション(文字列とフラグ)があります。文字列には任意の値を指定できますが、フラグはONかOFFのいずれかです。`sudoers`の設定ファイルで最も重要な構文構造は次のとおりです。

```
# Everything on a line after # is ignored ❶
Defaults !insults # Disable the insults flag ❷
Defaults env_keep += "DISPLAY HOME" # Add DISPLAY and HOME to env_keep ❸
tux ALL = NOPASSWD: /usr/bin/frobnicate, PASSWD: /usr/bin/journalctl ❹
```

- ❶ 例外が2つあります。`#include`と`#includedir`は通常のコマンドです。最近のバージョンでは`#`が使用されなくなっています。代わりに、`include`ディレクティブの前には`@`を記述します。下位互換性を維持するために、`#`表記が引き続きサポートされています。
- ❷ 目的のフラグをONに設定するには`!`文字を削除します。
- ❸ `env_reset`が有効な場合に保持する必要がある環境変数を列挙します。
- ❹ ユーザ`tux`は、`/usr/bin/journalctl`を実行するにはパスワードが必要だが、すべてのホスト上で`/usr/bin/frobnicate`を実行するにはパスワードは不要であることを規定する複雑なルール。

有用なフラグとオプション

targetpw

設定すると、`sudo`によって、`-u`オプションで指定されたユーザパスワードの入力が求められ、`-u`が使用されていない場合は`root`パスワードの入力が求められます。デフォルトはONです。

```
Defaults targetpw # Turn targetpw flag ON
```

rootpw

設定すると、`sudo`によって`root`パスワードの入力が求められます。デフォルトはOFFです。

```
Defaults !rootpw # Turn rootpw flag OFF
```

env_reset

設定すると、`sudo`は、`TERM`、`PATH`、`HOME`、`MAIL`、`SHELL`、`LOGNAME`、`USER`、`USERNAME`、および`SUDO_*`で構成される最小限の環境を構築します。また、`env_keep`に列挙されている変数は、呼び出し元の環境からインポートされます。デフォルトはONです。

```
Defaults env_reset # Turn env_reset flag ON
```

env_keep

`env_reset`フラグがONの場合に保持する環境変数の列挙。

```
# Set env_keep to contain EDITOR and PROMPT
Defaults env_keep = "EDITOR PROMPT"
```

```
Defaults env_keep += "JRE_HOME" # Add JRE_HOME
Defaults env_keep -= "JRE_HOME" # Remove JRE_HOME
```

env_delete

env_reset フラグがOFFの場合に削除する環境変数の列挙。

```
# Set env_delete to contain EDITOR and PROMPT
Defaults env_delete = "EDITOR PROMPT"
Defaults env_delete += "JRE_HOME" # Add JRE_HOME
Defaults env_delete -= "JRE_HOME" # Remove JRE_HOME
```

7.2 基本的なsudoersルール

各ルールは、次のスキームに従います([]はオプション部分を示しています)。

```
#Who      Where      As whom    Tag        What
User_List Host_List = [(User_List)] [NOPASSWD:|PASSWD:] Cmnd_List
```

SUDOERSルールの構文

User_List

1つ以上の(コンマで区切られた)識別子。ユーザ名、%GROUPNAME形式のグループ、または#UID形式のユーザIDを指定します。否定は!プレフィクスで指定できます。

Host_List

1つ以上の(コンマで区切られた)識別子。(完全修飾された)ホスト名またはIPアドレスのいずれかを指定します。否定は!プレフィクスで指定できます。通常、Host_ListにはALLを選択します。

NOPASSWD:|PASSWD:

NOPASSWD:の後に記述した、Cmnd_Listと一致するコマンドを実行する場合は、パスワードが要求されません。

PASSWD:はデフォルトです。PASSWD:とNOPASSWD:の両方が同じ行に存在する場合にのみ指定する必要があります。

```
tux ALL = PASSWD: /usr/bin/foo, NOPASSWD: /usr/bin/bar
```

Cmnd_List

1つまたは複数の(コンマで区切られた)指定子。実行可能ファイルへのパスの後に、オプションで使用可能な引数を指定します。

```
/usr/bin/foo # Anything allowed
```

```
/usr/bin/foo bar # Only "/usr/bin/foo bar" allowed
/usr/bin/foo "" # No arguments allowed
```

ALLは、User_List、Host_List、およびCmnd_Listとして使用できます。

7.3 エイリアスを使用したsudoersの簡潔化

エイリアスを取り入れて項目をグループ化すると、管理者は繰り返しの多い一連のルールを個別に維持する必要がなくなります。これらの構文はルールの構文と同じです。次のタイプのエイリアスがサポートされています。

User_Alias

ユーザ名のリスト

Runas_Alias

UID別のユーザのグループ

Host_Alias

ホスト名のリスト

Cmnd_Alias

コマンドとディレクトリ、およびエイリアスのリスト

エイリアスは、ユーザ、グループ、コマンド、およびホストの名前付きリストと考えてください。次にエイリアスの機能例を挙げます。

```
Host_Alias    WEBSERVERS = www1, www2, www3 ❶
User_Alias    ADMINS = tux, wilber, suzanne ❷
Cmnd_Alias    REBOOT = /sbin/halt, /sbin/reboot, /sbin/poweroff ❸
ADMINS WEBSERVERS = REBOOT ❹
```

- ❶ 3台のサーバを1つのHost_Alias WEBSERVERSにグループ化しています。(完全修飾)ホスト名またはIPアドレスを使用できます。
- ❷ 前記のグループ化したホストと同様に、グループユーザやユーザのグループ(%wheelなど)も、ここに列挙します。否定は、通常どおりにプレフィクス!!で指定します。
- ❸ 同じコンテキストで使用するコマンドのグループを指定しています。
- ❹ Host_Aliasで指定したすべてのホスト上でUser_Aliasで指定したすべてのユーザが、Cmnd_Aliasで指定したコマンドグループを実行できることを規定した単一のルールにすべてのエイリアスをまとめています。

要約すると、エイリアスは、管理者が `sudoers` を管理しやすく簡潔な状態(したがって安全な状態)に保つのに役立ちます。たとえば、ユーザの1人が退職した場合、この特定のユーザを含むルールをすべて検索するのではなく、`User_Alias`ステートメントと、そのユーザが属していたシステムグループからこの人物の名前を一度削除するだけです。

8 法的事項

Copyright © 2006–2025 SUSE LLC and contributors. All rights reserved.

この文書は、GNUフリー文書ライセンスのバージョン1.2または(オプションとして)バージョン1.3の条項に従って、複製、頒布、および/または改変が許可されています。ただし、この著作権表示およびライセンスは変更せずに記載すること。ライセンスバージョン1.2のコピーは、「GNUフリー文書ライセンス」セクションに含まれています。

SUSEの商標については、<https://www.suse.com/company/legal/> を参照してください。その他の第三者のすべての商標は、各社の所有に帰属します。商標記号(®、™など)は、SUSEおよび関連会社の商標を示します。アスタリスク(*)は、第三者の商標を示します。

本書のすべての情報は、細心の注意を払って編集されています。しかし、このことは正確性を完全に保証するものではありません。SUSE LLC、その関係者、著者、翻訳者のいずれも誤りまたはその結果に対して一切責任を負いかねます。

A GNUフリー文書利用許諾契約書(GFDL)

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. この使用許諾書を一字一句そのままの複製および頒布することは許可されますが、変更は許可されません。

0. 序文

この利用許諾契約書の目的は、マニュアル、テキストブック、またはその他の機能的で有用な文書を、自由という意味で「フリー」にすることです。つまり、そのような文書を、変更の有無や商用非商用に関わらず、コピーまたは再配布する実効的な自由をすべての人々に保証することです。第二に、本利用許諾契約書は、作者または発行者が他者によって行われた変更について責任を負わないとともに、その著作物の功績が確保されるように意図されています。

本利用許諾契約書は、「コピーレフト」(著作物を自由に複製および改変できるようにすること)の一種であり、文書の派生著作物は、それ自体が同じ意味においてフリーでなければなりません。フリーソフトウェア向けに考慮されたコピーレフト利用許諾であるGNU一般公衆利用許諾契約書(GPL)を補足するものです。

弊社は、この利用許諾契約書をフリーソフトウェアのマニュアルに使用するために設計しました。それは、フリーソフトウェアにはフリーマニュアルが必要であるためです。つまり、フリープログラムには、そのソフトウェアと同じ自由を提供するマニュアルが付属しなければなりません。ただし、本利用許諾契約書は、ソフトウェアマニュアルに制限されるものではありません。主題であるか否か、または印刷された本として発行されるか否かに関わらず、任意のテキスト著作物に使用することができます。本利用許諾契約書は、その目的が指示または参照に置かれている著作物に主に使用することを推奨します。

1. 適用範囲と定義

本利用許諾契約書は、この利用許諾の条項に従って頒布できることを定めた著作権者の通告が記載されている任意のメディアにおけるマニュアルまたは他の著作物に適用されます。そのような通告は、その著作物をここに記載されている条件に従って使用するための世界的な無償の利用許諾を無期限で付与します。次に示す「文書」は、そのような任意のマニュアルまたは著作物を指します。その公衆ユーザはいずれも被許諾者であり、「利用者」と呼ばれます。利用者は、著作権法に従った許可が必要になるような方法で著作物を複製、変更または頒布する場合に、利用許諾を受け入れます。

文書の「変更された版」とは、そのまま複製されるか、変更または別の言語に翻訳された(またはその両方)文書あるいはその一部を含んだ著作物のことです。

「二次セクション」は、文書の発行者または作者と文書の全体的な主題(または関連事項)との関係のみを示す文書の名前付き付録または前付け部分です。総体的な主題に直接関わる内容は含まれていません。(したがって、文書が部分的に数学のテキストブックになっている場合、二次セクションでは数学について説明されない場合があります)。関係には、主題または関連事項との歴史的なつながり、あるいはそれらに関する法的、商的、哲学的、倫理的、政治的位置付けが含まれる場合があります。

「不変セクション」は、文書が本利用許諾契約書の条件の下でリリースされる旨を述べている通告において、そのタイトルが不変セクションのものとして指定されている、ある特定の二次セクションです。セクションが、すでに説明した二次セクションの定義に一致しない場合は、不変として指定することはできません。文書には、不変セクションが含まれない場合があります。文書で不変セクションを特定しない場合、不変セクションは含まれません。

「カバーテキスト」とは、文書が本利用許諾契約書の条件の下でリリースされる旨を述べている通告において、表カバーテキストまたは裏カバーテキストとして列挙されている、あるいは一定の短い文章のことで、表カバーテキストは、最大で5語、裏カバーテキストは、最大で25語によって構成できます。

文書の「透過的な複製」とは、その仕様が一般の利用者にとって入手可能で、一般的なテキストエディタまたは一般的な描画プログラム(画素で構成される画像用)、あるいは広く使用されている図面エディタ(図面用)で文書を直接改訂するのに適した形式で表される機械可読の複製のことです。テキストフォーマットへの入力またはテキストフォーマットへの入力に適したさまざまな形式への変換に適していることも前提になります。読者による以後の変更を阻止または妨げるようにマークアップまたはマークアップのない状態が調整されている、他の点では透過的なファイル形式で行われた複製は、透過的な複製ではありません。イメージ形式は、相当量のテキストに使用されている場合、透過的ではありません。「透過的」ではない複製は、「不透明」と呼ばれます。

透過的な複製に適した形式として、マークアップのないプレーンなASCII、Texinfo入力形式、LaTeX入力形式、一般に取得可能なDTDを使用するSGMLまたはXML、標準に準拠したHTML、人為的変更用のPostScriptまたはPDFがあります。透過的なイメージ形式には、PNG、XCF、JPGが含まれます。不透明な形式には、独自のワードプロセッサのみで読み取りおよび編集を行える独自の形式、DTDまたは処理(またはその両方)ツールを一般に取得できないSGMLまたはXML、機械生成HTML、出力のみを目的として一部のワードプロセッサによって作成されるPostScriptまたはPDFが含まれます。

「タイトルページ」とは、印刷された本の場合、タイトルページ自体、および本利用許諾契約書でタイトルページに表示することが要求されるマテリアルを読みやすいように保持するために必要な以降のページのことを指します。そのようなタイトルページがない形式の著作物の場合、「タイトルページ」は、本文の開始部分に先行する、著作物のタイトルを最も顕著に表している部分の近くにあるテキストのことを指します。

「XYZという表題の付いた」セクションとは、そのタイトルが正確にXYZになっているか、またはXYZを別の言語に翻訳しているテキストに続いてカッコ付きのXYZが含まれている文書の名前付きサブユニットのことです。(ここで、XYZは、次に示すように、「謝辞」、「献辞」、「推薦」、「履歴」などの特定のセクション名を表します)。文書を変更するときに、そのようなセクションの「タイトルを保存する」とは、この定義に従って「XYZという表題の付いた」セクションが残されることを表します。

文書では、本利用許諾契約書が文書に適用される旨を述べている通告の付近に保証の放棄を含めることができます。保証の放棄条項は、本利用許諾契約書内の参照によって、保証の放棄に関してのみ組み込まれると見なされます。つまり、これらの保証の放棄条項がもつ可能性のある他のいかなる含意も無効であり、本利用許諾契約書の意味にまったく影響を与えません。

2. そのままの複製

利用者は、商用か否かを問わず、任意のメディアにおいて文書を複製または頒布することができます。その際に、本利用許諾契約書、著作権表示、および本利用許諾契約書が文書に適用される旨を述べる利用許諾通告をすべての複製で再生し、本利用許諾契約書の条件に他のいかなる条件も追加しないことが前提条件になります。利用者は、技術的手段によって、作成または頒布する複製の読み込みまたはさらなる複製を妨げたり、制御したりすることはできません。ただし、複製と引き換えに対価を受け取ることができます。十分に大量の複製を頒布する場合は、セクション3の条件に従う必要もあります。

すでに述べた同じ条件に従って複製を貸与したり、複製を公開したりすることもできます。

3. 大量の複製

発行する文書の印刷した複製(または、通常、印刷したカバーをもつメディアに含まれた複製)が100部を超え、文書の利用許諾通告でカバーテキストを必要とする場合は、すべてのカバーテキスト(表カバーの表カバーテキスト、裏カバーの裏カバーテキスト)を明瞭かつ読みやすく記載したカバーに文書の複製を同封する必要があります。また、両方のカバーでは、これらの複製の発行者として、利用者を読みやすい状態で明確に識別しなければなりません。表カバーには、フルタイトルを記述し、タイトルのすべての語が同等に目立つようにする必要があります。カバーには他のマテリアルを追加することもできます。カバーに限って変更を行った場合の複製は、文書のタイトルが保持されていて、これらの条件を満たしている限り、他の点に関してそのままの複製と見なすことができます。

いずれかのカバーで、必要なテキストが多すぎて、読みやすい状態に収まらない場合は、列挙されている最初の部分(問題なく収まる分)を実際のカバーに記載し、残りの部分を隣接ページに入れます。

文書の不透明な複製を100部以上公開または頒布する場合は、それぞれの不透明な複製とともに機械可読の透過的な複製を含めるか、それぞれの不透明な複製内あるいはその複製とともに、ネットワークの一般利用者が標準的な一般ネットワークプロトコルを使用して、追加マテリアルのない文書の完全な透過的な複製をダウンロードするときにアクセスできるコンピュータネットワークの場所を明記する必要があります。後者のオプションを使用する場合は、不透明な複製の大量頒布を開始するときに十分慎重な手順を取り、この透過的な複製が、その版の不透明な複製を最後に一般頒布した後(直接またはエージェントや小売業者を通じて)少なくとも1年間、指定した場所で継続的にアクセス可能となるように配慮する必要があります。

大量の複製を再頒布する時点よりもかなり前に、文書の作者に連絡して、文書の更新版を提供する機会を与えることが要求されますが、必須ではありません。

4. 変更

文書の変更された版を、すでに述べた第2項および第3項の条件に従って複製および頒布することができます。その際は、本利用許諾契約書に確実に従って、変更された版をリリースし、変更された版が文書の役割を担うようにして、その複製を所要する任意の利用者に変更された版の頒布および変更の利用許諾を与えることが前提になります。また、変更された版で次のことを行う必要があります。

- A. タイトルページ(カバーがある場合はカバー上も含める)で、文書、および以前の版の文書(以前の版がある場合は、その旨、文書の履歴セクションに列挙する)と識別されるタイトルを使用します。前の版と同じタイトルは、その版の元の発行者が許可を与えた場合に、使用することができます。
- B. タイトルページ上に、この要件から解放されない限り、変更された版において変更の著者としての責任を担う1人以上の人またはエンティティとともに、文書の筆頭著者を少なくとも5人、作者として列挙します(5人に満たない場合は、その筆頭著者のすべて)。
- C. タイトルページ上に、変更された版の発行者の名前を、発行者として記載します。
- D. 文書のすべての著作権表示を保持します。
- E. 変更に関する適切な著作権表示を、他の著作権表示の隣に追加します。
- F. 著作権表示の直後に、本利用許諾契約書の条項に従って変更された版を利用するための許可を一般利用者に与える利用許諾通告を、次の補遺に示す形式で含めます。
- G. その利用許諾通告に、不変セクションの全リスト、および文書の利用許諾通告で指定されている必須カバーテキストを保持します。
- H. 本利用許諾契約書の変更されていない複製を含めます。
- I. 「履歴」という表題のセクションを保持して、そのタイトルを保持し、タイトルページに記載されているとおりに、変更された版のタイトル、年度、新しい作者、発行者を少なくとも示す項目を追加します。文書に履歴というセクションがない場合は、そのタイトルページに記載されているとおりに文書のタイトル、年度、作者、発行者を示すセクションを作成し、前の文章に記載されているとおりに変更された版を示す項目を追加します。
- J. 文書の透過的な複製に一般利用者がアクセスできるように文書で指定されている場合は、そのネットワークの場所、およびその文書の基盤となった前の版に対応して文書で指定されているネットワークの場所を保持します。これらは、「履歴」セクションに配置することができます。文書自体よりも4年以上前に発行された著作物の場合、または参照されているその版の元の発行者が許可を与えている場合は、そのネットワークの場所を省略することができます。

- K. 「謝辞」または「献辞」という表題のセクションの場合は、そのセクションのタイトルを保持し、セクション内に、それぞれの貢献者謝辞またはその中の献辞(またはその両方)のすべての内容と意味合いを保持します。
- L. 文書のすべての不変セクションを保持し、そのテキストおよびタイトルを未変更のままにします。セクション番号またはそれと同等の要素は、セクションタイトルの一部と見なされません。
- M. 「推薦」という表題の任意のセクションを削除します。そのようなセクションは、変更された版に含めることはできません。
- N. 既存のセクションのタイトルを変更して、「推薦」という表題にしたり、タイトルが不変セクションと矛盾したりしないようにします。
- O. 保証の放棄を保持します。

変更された版に、二次セクションと見なされ、文書から複製されたマテリアルを含まない新しい前付けセクションまたは付録が含まれる場合は、これらの一部またはすべてを任意に「不変」として指定することができます。これを行うには、変更された版の利用許諾表示内で列挙されている不変セクションにそのタイトルを追加します。これらのタイトルは、他のすべてのセクションタイトルと異なっている必要があります。

「推薦」という表題のセクションを追加することができますが、その際は、変更された版のさまざまな当事者による推薦以外の要素が含まれていないことが前提になります。たとえば、校正者によるコメント、または文が標準的な信頼できる定義として組織によって承認されていることを示すという宣言文などが相当します。

表カバーテキストとしての最大5語の短い文、および裏カバーテキストとしての最大25語の短い文を、変更された版のカバーテキストのリストの終わりに追加できます。表カバーテキストの短い1文および裏カバーテキストの短い1文のみを、1つのエンティティが追加できます(またはエンティティによって行われた調整を通じて)。文書に、利用者または利用者が関わる同じエンティティによって行われた調整を通じて前に追加された同じカバーのカバーテキストがすでに含まれている場合は、別のものを追加することはできませんが、古いものを置き換えることができます。ただし、その古いものを追加した前の発行者から明示的な許可を得る必要があります。

文書の作者および発行者は、本利用許諾契約書により、その名前を得るために使用したり、変更された版の推薦を主張または暗示したりする許可を与えるものではありません。

5. 文書の結合

文書は、すでに述べた変更された版に関するセクション4の条件に従って、本利用許諾契約書の下でリリースされた他の文書と結合することができます。その際は、その組み合わせの中に、元の全文書のすべての不変セクションを未変更のまま含めて、そのすべてを結合された著作物の不変セクションとしてその利用許諾表示に列挙し、そのすべての保証の放棄を保持することが前提となります。

結合された著作物には、本利用許諾契約書の複製を1つのみ含める必要があります。複数の同一の不変セクションは、単一の複製で置き換えることができます。同じ名前だが内容の異なる複数の不変セクションがある場合は、そのような各セクションのタイトルを固有なものにします。その際は、その終わりに、カッコ付きで、そのセクションの元の作者または発行者の名前(既知の場合)、あるいは固有の番号を追加します。不変セクションのリスト内のセクションタイトルには、結合された著作物の利用許諾表示の場合と同じ調整を加えます。

組み合わせでは、さまざまな元の文書の「履歴」という表題のセクションを結合して、1つの「履歴」というセクションを構築する必要があります。同じように、「謝辞」という表題のセクション、および「献辞」という表題のセクションも結合します。「推薦」という表題のすべてのセクションを削除する必要があります。

6. 文書のコレクション

文書および本利用許諾契約書の下でリリースされた他の文書から成るコレクションを作成して、さまざまな文書に含まれる本利用許諾契約書の個々の複製を、コレクションに含まれる単一の複製で置き換えることができますが、他のすべての点での各文書のそのままの複製に関する本利用許諾契約書の規則に従うことが前提になります。

そのようなコレクションから単一の文書を抽出して、その文書の本利用許諾契約書に従って個々に頒布することができますが、その際は、本利用許諾契約書の複製を抽出した文書に挿入して、その文書のそのままの複製に関するその他のすべての点で、本利用許諾契約書に従う必要があります。

7. 独立した著作物の集積

文書またはその派生物を他の個別および独立した文書または著作物とともに、ストレージまたは頒布メディア内またはそのボリューム上に蓄積することを「集積」と呼びます。その場合は、個々の著作物の許可を超えてその蓄積の利用者の法的権限を制限することに、蓄積による著作権を使用しないことが前提になります。文書が集積に含まれる場合、本利用許諾契約書は、それ自体が文書の派生著作物ではない集積内の他の著作物に適用されません。

セクション3のカバーテキスト要件が文書のこれらの複製に適用可能であり、文書が集積全体の半分に満たない場合は、文書のカバーテキストを、集積内の文書のカバー、または文書が電子形式の場合は、電子的な同等のカバーに配置することができます。それ以外の場合は、集積全体の印刷されたカバー上に表示する必要があります。

8. 翻訳

翻訳は一種の変更と見なされるため、セクション4の条件に従って文書の翻訳を配布することができます。不変セクションを翻訳で置き換えるには、著作権者からの特別な許可が必要ですが、これらの不変セクションの元のバージョンのほかに、一部またはすべての不変セクションの翻訳を含めることができます。本利用許諾契約書、文書内のすべての利用許諾表示、および保証の放棄の翻訳を含めることができますが、その際は、本利用許諾契約書の元の英語版、およびそれらの利用許諾表示と保証の放棄の元の版も含めることが前提になります。本利用許諾契約書、利用許諾表示、または保証の放棄の翻訳と元の版との間に不一致がある場合は、元の版が優先されます。

文書内のセクションに、「謝辞」、「献辞」、または「履歴」という表題が付いている場合は、そのタイトルを保持する(セクション1)ための要件(セクション4)により、通常は実際のタイトルを変更する必要があります。

9. 終了

利用者は、本利用許諾契約書に明示的に記載されている形態を除き、文書を複製、改変、二次利用許諾、および頒布してはなりません。別の方法で文書を複製、改変、二次利用許諾、または頒布しようとするのは無効であり、本利用許諾契約書の下で利用者の権利は自動的に消滅します。ただし、本使用許諾の下で利用者から複製物または権利を受領した関係者は、条項を遵守している限り、権利が消滅することはありません。

10. 本利用許諾契約書の将来の改訂

フリーソフトウェア財団は、GNUフリー文書利用許諾契約書(GFDL)の新版または改訂版を随時公表することがあります。そのような新版は、性格的には現行版と似たものになりますが、新たな問題や懸案事項に対応するために細部が異なる可能性があります。<https://www.gnu.org/copyleft/>を参照してください。

本利用許諾契約書の各版には、区別するための版番号が設定されます。文書に、それに適用される本利用許諾契約書の特定の版番号と「後継版」が指定されている場合、利用者は、選択によって、その指定された版の条項またはフリーソフトウェア財団から公開される後継版の条

項(ドラフトではない)に従うことになります。文書に、本利用許諾契約書の版番号が指定されていない場合、利用者は、フリーソフトウェア財団からこれまでに公開された任意の版(ドラフトではない)を選択することができます。

補遺: 本利用許諾書をご使用の文書に使用する方法

```
Copyright (c) YEAR YOUR NAME.  
Permission is granted to copy, distribute and/or modify this document  
under the terms of the GNU Free Documentation License, Version 1.2  
or any later version published by the Free Software Foundation;  
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.  
A copy of the license is included in the section entitled "GNU  
Free Documentation License".
```

不変セクション、表カバーテキスト、および裏カバーテキストがある場合は、「with...Texts」の行を次のように置き換えます。

```
with the Invariant Sections being LIST THEIR TITLES, with the  
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

カバーテキストのない不変セクションが含まれている場合、またはこの3つの他の組み合わせの場合は、その2つの代替要素をマージして状況に合わせます。

文書にプログラムコードの重要な例が含まれている場合は、GNU一般公衆利用許諾契約書 (GPL)などの選択したフリーソフトウェアの利用許諾に従って、これらの例を平行してリリースし、フリーソフトウェアでのその利用を許可することを推奨します。