


# OBSKURA

あなたのデータに見えないセキュリティを

## ユーザーマニュアル

バージョン **0.3.0**

MACOS · WINDOWS · LINUX

2026 年 10 月

ユーザーマニュアル

# 目次


| | | | |
|---|---|---|---|
| 01 | Obskura へようこそ | できること、データベースと金庫 | 03 |
| 02 | インストールと初回起動 | macOS、Windows、Linux · アクセスキー · 画面構成 | 04 |
| 03 | パスワードとキーファイル | 認証情報、Argon2id、暗号方式の選択 | 05 |
| 04 | パスワードデータベース | 作成、ロック解除、エントリの整理 | 07 |
| 05 | エントリの管理 | 生成ツール、フィールド、添付ファイル、履歴 | 08 |
| 06 | データベースのツール | 検索、安全なコピー、データベースのメニュー | 09 |
| 07 | ファイル金庫 | 金庫を作ってファイルを暗号化する | 10 |
| 08 | ファイルを扱う | 開く、編集する、エクスポートする | 11 |
| 09 | リモートサーバー | SFTP、WebDAV、SMB、FTP/FTPS | 12 |
| 10 | 複数デバイスでリアルタイムに | リアルタイム同期、通知 | 14 |
| 11 | アクセスキーとデバイス | 付属ファイル、各デバイスの ID とパス | 15 |
| 12 | 自動バックアップ | 2 つ目の保存先、日付付きコピー、デバイスごとの設定 | 17 |
| 13 | 読み込みとエクスポート | オンラインのパスワード管理、KeePass などの CSV | 18 |
| 14 | 言語 | 自動の言語、XML ファイル、新しい翻訳 | 20 |
| 15 | ロックと設定 | 自動ロック、クリップボード、テーマ、デバイス名 | 23 |
| 16 | ショートカットと暗号 | キーボードショートカットと技術的な詳細 | 24 |
| 17 | 推奨事項とトラブルシューティング | バックアップ、よくある質問 | 25 |


**表記について**

ボタンやメニューの名前は**太字**で示します。キーの組み合わせは `Ctrl/⌘` + `L` のように表記し、`Ctrl` は Windows と Linux、`⌘` は macOS に対応します。ファイル名と拡張子は`等幅フォント`で示します。スクリーンショットはこのマニュアルの言語の画面です。

はじめに

# 01 OBSKURA へようこそ

Obskura は軍事レベルの暗号化でパスワードとファイルを守ります。アカウントも第三者のサービスも不要。データはあなたが決めた場所——コンピュータやあなたのサーバー——にとどまります。

データベースも金庫も、普通のファイルまたはフォルダです。コンピュータのディスク、USB メモリ、Dropbox・OneDrive・iCloud Drive などの同期フォルダ、あるいはご自身の SFTP・WebDAV・SMB・FTP サーバーに直接置くことができ、いくつでも作れます。個人用、仕事用、家族と共有するもの。複数のコンピュータから同時に使え、別々のパスでアクセスしていても互いの変更がリアルタイムで反映されます。

**パスワードデータベース**

KeePass をモデルにした 1 つの `.obk` ファイル。ユーザー名、パスワード、アドレス、メモ、カスタムフィールド、添付ファイル、変更履歴を持つエントリをグループに整理して保存します。

**ファイル金庫**

Cryptomator をモデルにしたフォルダ。書類や写真、アーカイブをドラッグすると、名前やフォルダ構造も含めてファイルが 1 つずつ暗号化されます。

## 保護のしくみ

データベースや金庫を作成するときは、**マスターパスワード**、**キーファイル**、またはその両方を選びます。Obskura はこの認証情報から Argon2id で暗号鍵を導出します。Argon2id はパスワードの推測を極端に遅くするために設計されたアルゴリズムです。その後、データは**TOP SECRET**（機密）指定の情報に承認された標準である AES-256 で暗号化されます。

認証情報はどこにも保存されません。認証情報がなければ、開発者を含め誰も内容を読めません。

**重要**

忘れたパスワードや失くしたキーファイルを復元する方法はありません。覚えられるパスワードを選び、キーファイルのコピーを安全な別の場所に保管してください。

## 概要

AES-256-GCM
XChaCha20-Poly1305
Serpent · Twofish · Camellia
暗号のカスケード
Argon2id 最大 2 GiB
HKDF-SHA256
ファイル名に AES-SIV
終了時にメモリを消去
アカウント不要
SFTP · WebDAV · SMB · FTPS
リアルタイム
自動バックアップ
デバイスごとのパス
8 言語
macOS · Windows · Linux

はじめよう

# 02 インストールと初回起動

### macOS

`.dmg` ファイルを開き、Obskura を**アプリケーション**にドラッグします。初回起動時に macOS がアプリをブロックした場合は、**システム設定** → **プライバシーとセキュリティ**で**このまま開く**を選びます。

### Windows

Windows の言語を自動で使う `.exe` インストーラーを実行するか、インストール不要のポータブル版を起動します。SmartScreen の警告が表示されたら、**詳細情報** → **実行**をクリックします。

### Linux

`.deb` パッケージ（Debian、Ubuntu）、`.rpm` パッケージ（Fedora、openSUSE）、またはインストール不要の `.AppImage` を使います。AppImage は実行権限を付けて起動します。

## 初回起動：言語とアクセスキー

Obskura はシステムの言語（イタリア語、英語、ドイツ語、スペイン語、ロシア語、中国語、日本語、アラビア語。それ以外は英語）で起動します。初回起動時に **Obskura アクセスキー**の作成を求められます。すでに別のコンピュータで Obskura を使っている場合は、そこで生成したキーを入力します。用途は第 11 章で説明します。

## スタート画面




図 1　サイドバーのあるようこそ画面。

**サイドバー**には**パスワードデータベース**と**ファイル金庫**のセクションがあります。各見出しの横には 2 つのボタンがあり、フォルダは既存の項目を**開く**ため、**＋** は新規作成のためのものです。

- 最近開いた項目は名前だけがリストに残り、パスは決して表示されません。錠前のアイコンは状態を示し、**閉じている**ときはロック中、**開いている**ときはロック解除中です。
- 項目にポインタを重ねると、最近使った項目から外す **×**（ファイルは削除されません）や、すぐにロックする錠前が表示されます。
- 下部：**すべてロック**と**設定**の歯車。

データを守る

# 03 パスワードとキーファイル

データベースと金庫は、**マスターパスワード**、**キーファイル**、または両方で開きます。両方を組み合わせるのが最も強力な保護です。あなたが*知っているもの*と、*持っているもの*の両方が必要になります。

## マスターパスワード

複雑な単語より長いフレーズのほうが安全です。ランダムな 4〜5 語は覚えやすく、推測は非常に困難です。入力欄の下のメーターが強度をビットで示します。

ランダムなパスワードがよければ、サイコロのボタンで**生成ツール**を開きます。

## キーファイル

KeePass と同じく、**どんなファイルでも**キーにできます。**既存のファイルを使う**か**新しく生成**を選べます。後者では Obskura が指定した場所にランダムなキーファイルを作成します。

たとえば USB メモリに入れ、データベースとは別に保管します。Obskura は各デバイスでの場所を記憶し、開くたびに自動で探します（第 11 章）。

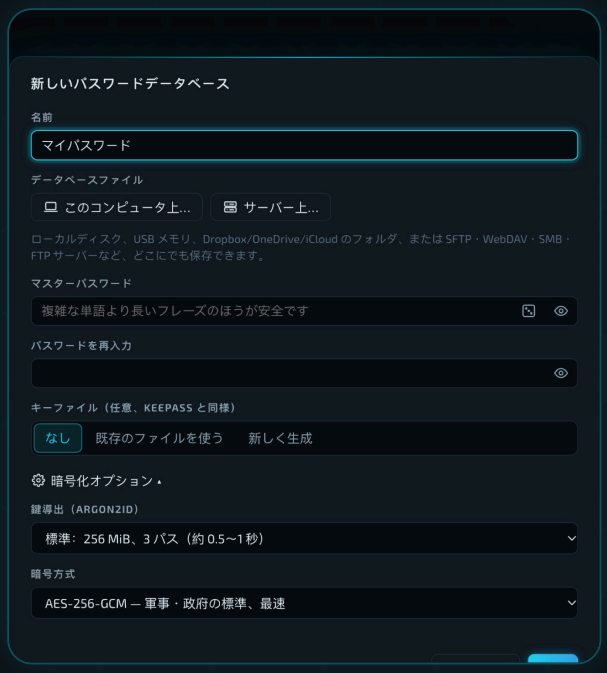



**図 2** 暗号化オプションを開いた状態のデータベース作成画面。

## 暗号化オプション

**暗号化オプション**では、鍵導出のコストと使用する暗号方式を選びます。既定の「標準」と AES-256-GCM で、どんな用途にも十分安全です。ほかの選択肢は次のページで説明します。

> **キーファイルは絶対に変更しない**
>
> 任意のファイル（写真や書類）をキーにした場合、ごくわずかな変更でも使えなくなります。Obskura が生成するファイルは認識可能なテキスト形式で、改行コードが変わっても有効です。いずれにしてもバックアップを保管してください。

データを守る

# 03 暗号方式の選び方

## 鍵導出（ARGON2ID）

Argon2id は認証情報を暗号鍵に変換します。使うメモリとパス数が多いほど、攻撃者が何百万ものパスワードを試すコストは高くなります。

| レベル | パラメータ | 開くまでの時間 |
|---|---|---|
| 標準 | メモリ 256 MiB、3 パス | 約 0.5〜1 秒 |
| 高 | 1 GiB、4 パス | 約 2〜4 秒 |
| 最高 | 2 GiB、6 パス | 約 5〜12 秒 |

**最高レベル**

開くたびに **2 GiB の空きメモリ**が必要です。メモリの少ないコンピュータではデータベースや金庫を開けない場合があります。十分な性能のコンピュータで使う場合にだけ選んでください。

## 暗号方式

すべての暗号方式は 256 ビットの鍵を使い、**認証付き**です。データのどんな改ざんも検出されます。

| 暗号方式 | 特徴 |
|---|---|
| AES-256-GCM | **既定。**TOP SECRET 情報に承認された米国の標準。ハードウェア支援により最も高速です。 |
| XChaCha20-Poly1305 | ストリーム暗号。AES 命令のないプロセッサでも高速です。 |
| Serpent-256-GCM | AES コンペの最終候補で、安全余裕が最も大きい（32 ラウンド）。やや低速です。 |
| Twofish-256-GCM | AES コンペの最終候補。Bruce Schneier らが設計。 |
| Camellia-256-GCM | 国際標準 ISO/IEC。欧州（NESSIE）と日本（CRYPTREC）で推奨。 |
| カスケード AES · Twofish · Serpent | **最も強力な選択。**独立した鍵で 3 回続けて暗号化します。将来 3 つのうち 2 つのアルゴリズムが破られてもデータは守られます。 |
| カスケード AES · XChaCha20 | まったく異なる系統のブロック暗号とストリーム暗号を組み合わせます。 |

**金庫**

金庫では、選んだ暗号方式が鍵と各ファイルの内容の両方を保護します。**パスワード / キーファイルを変更**で変更した場合、既存のファイルは以前の方式のまま、以降に追加したファイルは新しい方式で暗号化されます。使用中の方式は金庫の右下に表示されます。

パスワードマネージャー

# 04 パスワードデータベース

## データベースを作成する

1. スタート画面で**新しいデータベース**をクリックするか、**パスワードデータベース**の横の **+** をクリックします。
2. **名前**を入力し、`.obk` ファイルの保存先として**このコンピュータ上…**か**サーバー上…**を選びます。
3. パスワードとキーファイルの一方または両方を設定します（第 03 章）。
4. **作成**を押します。データベースはロック解除された状態で開きます。

## 開いてロックを解除する

1. サイドバーでデータベースを選ぶか、フォルダアイコンで既存の `.obk` ファイルを開きます。
2. パスワードを入力します。キーファイルとバックアップコピーは自動で探されます。キーファイルが見つからない場合は**選択…**で選びます。
3. **ロック解除**を押すか `Enter` キーを押します。




図 3　ロックを解除したデータベース：グループ、エントリ一覧、選択したエントリの詳細。

## 3 つの列

- **グループ**：**すべてのエントリ**、**お気に入り**、作成したフォルダ。グループにポインタを重ねると、サブグループの作成・編集・削除のコマンドが表示されます。**新しいグループ**はリストの最後にあります。
- **エントリ**：各エントリのタイトルとユーザー名。星はお気に入り、クリップは添付ファイルを示します。**ダブルクリック**でパスワードを直接コピーできます。
- **詳細**：エントリのすべてのフィールドと、表示・コピー・ブラウザでアドレスを開くボタン。

パスワードデータベース

# 05 エントリの管理

エントリを追加するには**新しいエントリ**（または `Ctrl/⌘` + `N`）を押します。変更するにはエントリを選んで**編集**を押します。下のボタンか `Ctrl/⌘` + `S` で保存します。変更はすぐにデータベースファイルに書き込まれます。




図 4 生成ツールを開いたエントリエディタ。

## エントリのフィールド

| | |
|---|---|
| **タイトルとグループ** | エントリの名前と、入っているフォルダ。 |
| **ユーザー名** | アカウント、メールアドレス、顧客番号。 |
| **パスワード** | 強度メーターと生成ツール付き。 |
| **アドレス** | 詳細画面からブラウザで開けます。 |
| **メモ** | 自由なテキスト。ほかの情報と同じく暗号化されます。 |
| **タグ** | カンマ区切りのキーワード。 |
| **有効期限** | この日を過ぎたらパスワードを変更する目安。 |

上部の**星**でエントリを**お気に入り**に追加します。

## パスワード生成ツール

パスワード欄のサイコロをクリックします。**長さ**（8〜64 文字）と文字の種類（`a-z`、`A-Z`、`0-9`、`!@#$`）を選びます。**紛らわしい文字を除く**で l、1、O、0 のような紛らわしい文字を避けられます。その後**再生成**、**コピー**、**このパスワードを使う**のいずれかを選びます。パスワードはシステムの暗号論的乱数生成器で作られます。

## カスタムフィールド

PIN、リカバリーコード、秘密の質問などの追加情報を入れられます。**保護**にチェックすると、パスワードと同様に伏せて表示されます。

## 添付ファイルと履歴

1 つあたり最大 **64 MB** のファイルを添付できます（それより大きいファイルは金庫を使ってください）。変更のたびに以前の版が残り、**履歴**は直近 **20** 件を保持して古いパスワードもコピーできます。

パスワードデータベース

# 06 データベースのツール

## 検索

上部の**データベース全体を検索**欄（または `Ctrl/⌘` + `F`）は、入力に合わせてすべてのグループのエントリを絞り込みます。`Esc` で検索をクリアします。

## 安全なコピー

ユーザー名、パスワード、保護フィールドの横のコピーボタンは値をクリップボードに入れ、Obskura が 20 秒後（設定で変更可）に**自動で消去**します。消去されるのはコピーした値がまだ残っている場合だけで、その間に別のものをコピーした場合はそのままです。

## データベースのメニュー

右上の ⋯ ボタンには次の項目があります：

| 項目 | 用途 |
|---|---|
| **データベースのプロパティ** | データベースの名前と説明を変更します。 |
| **パスワード / キーファイルを変更** | 新しい認証情報を設定します。データベースは新しい鍵で暗号化し直されます。 |
| **バックアップコピーを保存…** | 暗号化したデータベースのコピーを別の場所に書き出します。 |
| **自動バックアップ…** | コピーの 2 つ目の保存先を設定します（第 12 章）。 |
| **デバイスとパス…** | データベースを使っているデバイスとそのパスを確認します（第 11 章）。 |
| **読み込み / エクスポート** | オンラインのパスワード管理や CSV からの読み込み、CSV へのエクスポート（第 13 章）。 |
| **データベースをロック** | データベースを閉じ、メモリからデータを消去します。 |

## 保存と同期

変更はすぐに**アトミック**に保存されます。新しいファイルは別に書き込まれ、完成してから古いファイルと置き換わるので、停電でも壊れません。その間に別のコンピュータがデータベースを変更していた場合、Obskura は保存前に 2 つの版を**統合**するため、何も失われません（第 10 章）。判断が必要なのは、別のデバイスで**パスワードまたはキーファイル**が変更された場合だけです。その場合 Obskura は、データベースをロックして新しい認証情報で開き直すよう提案します。

> **ヒント**
> 同じデータベースを複数のコンピュータで使うには、ご自身のサーバー（第 09 章）か同期フォルダに置きます。どこでも同時に開いたままにできます。

暗号化されたファイル

# 07 ファイル金庫

金庫とは、ファイルを暗号化した状態で収めるフォルダです。外から見ると読めない名前のファイルが並んでいるだけで、中身も整理のされ方もわかりません。

## 金庫を作成する

1. **新しい金庫**をクリックするか、**ファイル金庫**の横の **+** をクリックします。
2. 名前と作成先のフォルダを選びます。Dropbox、OneDrive、iCloud Drive の中やサーバー上でもかまいません。
3. パスワードとキーファイルの一方または両方を設定し、**作成**を押します。

## ファイルを追加する

- Finder / エクスプローラーからファイルやフォルダをウィンドウに**ドラッグ**すると、表示中のフォルダに暗号化されて入ります。
- または**ファイルを追加**か**フォルダを追加**を使います。
- 元のファイルはそのままです。暗号化したあと、ご自身で削除できます。

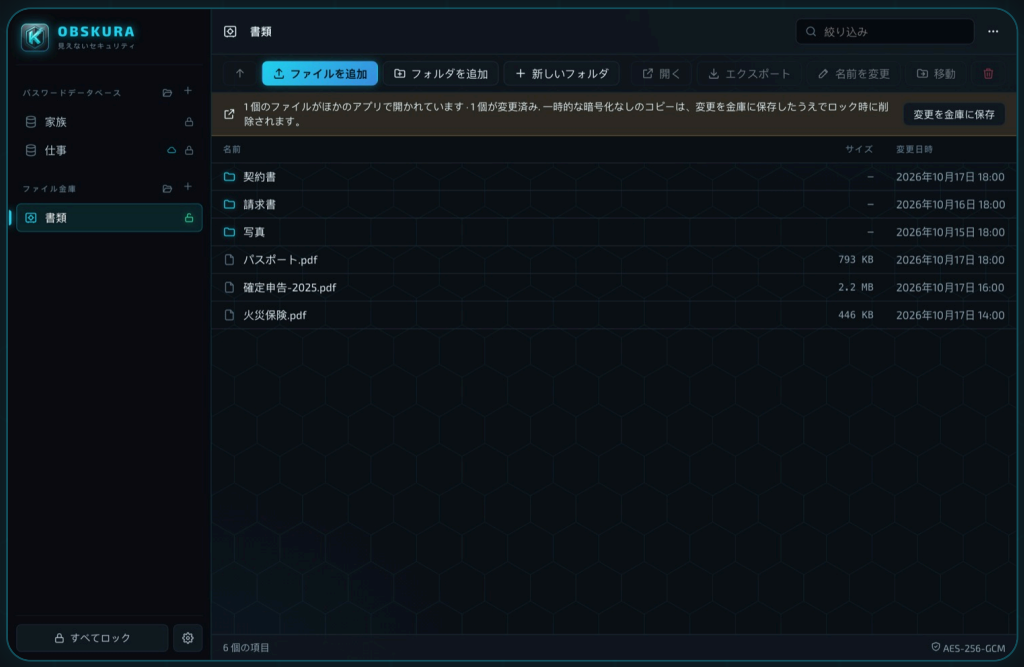



図 5 ロックを解除した金庫。

## コマンドバー

↑ で上のフォルダへ · **新しいフォルダ**でサブフォルダを作成 · **開く**、**エクスポート**、**名前を変更**、**移動**、**削除**は選択した項目に作用します。`Ctrl/⌘` + クリックや `Shift` + クリックで複数選択できます。下部のステータスバーには項目数と処理の進み具合が表示されます。

ファイル金庫

# 08 ファイルを扱う

## ファイルを開いて編集する

ファイルをダブルクリックします（または選んで**開く**を押します）。Obskura は保護された一時フォルダに復号し、システムの既定のアプリで開きます。

1. もう一方のアプリで普段どおり編集して保存します。
2. Obskura に、開いているファイルと変更されたファイルを示すバナーが表示されます。
3. **変更を金庫に保存**を押して、新しい版を暗号化し直します。

> **一時的な平文のコピー**
> ファイルを開いている間は、復号したコピーがディスク上に存在します。金庫をロックすると、Obskura は残っている変更を先に保存してから**一時コピーを削除**します。ロックする前に、もう一方のアプリでファイルを閉じてください。

## エクスポート

**エクスポート**は選んだ項目をフォルダごと、指定したフォルダに復号します。エクスポートしたファイルは**暗号化されていません**。金庫の外で必要なときだけ使ってください。

## 整理

名前の変更は**名前を変更**か `F2`、移動は**移動**で移動先フォルダを選びます。**削除**（または `Delete`）は必ず確認を求めます。**絞り込み**欄は、入力した文字を名前に含む項目だけを表示します。

## クラウドとの同期

金庫内のファイルは 1 つずつ暗号化されているので、書類を 1 つ変更すると同期サービスはそのファイルだけをアップロードします。そのため金庫は Dropbox、OneDrive、Google Drive、iCloud Drive の中でもうまく動きます。

## ディスク上の金庫の構成

| 項目 | 内容 |
|---|---|
| `vault.obkv` | 金庫の鍵。あなたの認証情報で暗号化されています。 |
| `vault.obkv.bak` | 最初のファイルが壊れた場合に備えた同一の予備コピー。 |
| `vault.obkx` | アクセスキーで暗号化された付属ファイル：キーファイル、バックアップ、各デバイスのパス（第 11 章）。 |
| `d/` | 暗号化されたすべてのファイルとフォルダが並んでいます。実際の構造は見えません。 |

既存の金庫を開くには、**ファイル金庫**の横のフォルダアイコンを使い、金庫のメインフォルダを選びます。中のファイルを手で変更したり移動したりしないでください。

保存先

# 09 リモートサーバー

データベースや金庫はコンピュータ上になくてもかまいません。Obskura は同期ソフトを介さず、ご自身のサーバー上で直接開いて保存します。データは**暗号化された状態で**送受信・保存されるため、サーバーの管理者や侵入者には読めないファイルしか見えません。

| プロトコル | 使われる場所 | 備考 |
|---|---|---|
| SFTP | NAS（Synology、QNAP、TrueNAS）、Linux サーバー、VPS | 推奨。パスワードまたは SSH 鍵。 |
| WebDAV | Nextcloud、ownCloud、Synology、QNAP、Box、Koofr | HTTPS 経由。Nextcloud のパスは `/remote.php/dav/files/<ユーザー>`。 |
| SMB | Windows の共有フォルダ、NAS、Samba | SMB 2 と 3。共有名を指定します。 |
| FTPS | TLS 対応のホスティングと FTP サーバー | `AUTH TLS` で暗号化した FTP。 |
| FTP | 従来の FTP サーバー | サーバーの認証情報が平文で送られます。FTPS か SFTP をおすすめします。 |

## 接続を設定する

1. **設定** → **サーバー接続**を開き、**新しい接続**を押します。
2. プロトコルを選び、アドレス、ポート（標準以外の場合）、ユーザー、パスワードを入力します。
3. **接続テスト**を押し、続けて**保存**を押します。

サーバーのパスワードは**システムのキーチェーン**（macOS のキーチェーン、Windows の資格情報マネージャー、Linux の Secret Service）に保存され、設定ファイルには決して書かれません。さらに暗号化して付属ファイルにもコピーされるので、ほかのコンピュータは自動で接続を再現します。




図 6 新しい SFTP 接続。

**サーバーキー（SFTP）**

初めて接続するとき、Obskura は `ssh` と同じようにサーバーキーの**フィンガープリント**を表示し、承認を求めます。あとでフィンガープリントが変わると、再度確認するまで接続は拒否されます。誰かがサーバーになりすましている可能性があるためです。サーバー側で何も変更していなければ承認しないでください。

### サーバー上で開いて保存する

- **開く：パスワードデータベース**や**ファイル金庫**の横のフォルダアイコンから**このコンピュータから…**と**サーバーから…**を選べます。スタート画面の**サーバーから…**も同じです。
- **作成：新しいデータベース**または**新しい金庫**ウィンドウで、**このコンピュータ上…**の代わりに**サーバー上…**を選びます。
- サーバー上の項目は、最近使った項目に**雲**のアイコン付きで表示されます。開くと上部バーに サーバー ラベルが表示されます。




図 7　サーバーブラウザー：接続とフォルダを選ぶ。

ブラウザーの左上で接続を選び、フォルダを移動します。データベースの場合は `.obk` ファイルをクリックし、金庫の場合はそのフォルダ（`vault.obkv` があるフォルダ）に入って**この金庫を開く**を押します。「フォルダ+」アイコンでサーバー上に新しいフォルダを作れます。

**パフォーマンス**

データベースは小さいので、リモートでも一瞬で開きます。金庫はファイルを開いたりエクスポートしたりするときに 1 つずつ転送するため、非常に大きなファイルでは回線速度が影響します。

共同作業

# 10 複数デバイスでリアルタイムに

同じデータベースや金庫を複数のコンピュータで同時に開けます。1 台で行った変更は、数秒でほかのコンピュータに表示されます。




図 8 通知：別のデバイスがエントリを追加・変更しました。

## しくみ

- Obskura はローカルディスクや同期フォルダでは **2 秒**ごと、サーバー上では **5 秒**ごとにファイルを確認します。確認で読むのはファイルの先頭 88 バイトだけです。
- 新しい版が見つかると、エントリ単位であなたの版と**統合**し、追加・変更・削除されたエントリを通知で示して、すぐに一覧を更新します。
- 2 人が**同じエントリ**を変更した場合は、新しいほうの変更が採用されます。もう一方は失われず、そのエントリの**履歴**に残ります。
- 削除は全員に反映されます。ただし、その間に誰かがそのエントリを変更していた場合は変更が優先され、エントリは残ります。
- 金庫では、あなたが表示しているフォルダで誰かがファイルを追加・名前変更・削除すると、一覧が自動で更新されます。

**コンピュータの時計**

どちらの変更が新しいかを判断するために、Obskura はコンピュータの時刻を使います。時刻を同期させておいてください（自動設定は通常どのシステムでも有効です）。

どこからでもすべて見つかる

# 11 アクセスキーとデバイス

各データベースと金庫の隣には小さな暗号化ファイルがあり、キーファイル・バックアップ・サーバーの場所と、各デバイスがそこへたどり着くパスを記録しています。これを守るのが Obskura アクセスキーです。

## OBSKURA アクセスキー

初回起動時に生成される（または別のコンピュータで作成したものを入力する）55 文字のコードです。システムのキーチェーンに保管され、**設定** → **Obskura アクセスキー**から次の操作ができます：

- **表示**：別のコンピュータで入力するために表示・コピーします。
- **キーを変更…**：新しいキーを生成します。Obskura はアクセスできる付属ファイルをすぐに暗号化し直し、それ以外は次に開いたときに更新します（以前のキーは読み取り用に引き続き有効です）。
- **キーを入力…**：別のコンピュータでキーが変更された場合に現在のキーを使います。

## 付属ファイル

AES-256-GCM で暗号化されており、アクセスキーがなければ読めません：

| ファイル | 内容 |
|---|---|
| `<名前>.obk.obkx` | 各データベースの隣：キーファイルの場所、バックアップの設定、サーバー接続、デバイスの記録。 |
| `vault.obkx` | 各金庫の中：同じ情報。 |
| `recenti.obkx` | アプリのデータフォルダ内：最近使った項目の一覧。ディスク上に平文のパスは残りません。 |

## 別のコンピュータから開く

新しいコンピュータでアクセスキーを入力すれば、データベースを選ぶだけです。Obskura は付属ファイルを読み、足りないサーバー接続を再現し、キーファイルを見つけ（そのデバイスで既知の場所、データベースの隣、ほかのデバイスのパス、バックアップコピーの順に探します）、元のファイルにアクセスできない場合はバックアップコピーを提案します。

**重要**

アクセスキーはデータベースのパスワードの代わりにはなりません。パスワード（とキーファイル）がなければデータは読めないままです。それでも安全な場所に保管してください。アクセスキーがないと、キーファイルとバックアップを手動で選ぶ必要があります。

## デバイスとパス

Obskura を使うコンピュータにはそれぞれ、初回起動時に生成される**一意の ID**と**名前**（コンピュータ名。**設定** → **このデバイスの名前**で変更可）があります。データベースや金庫を開くたびに、暗号化された付属ファイルに自分の記録を書き込みます：

| 項目 | 例 |
|---|---|
| **ID と名前** | `3f2a9c1e-…` ·「自宅の MacBook」 |
| **システムとバージョン** | macOS · Obskura 0.3.0 · 最終アクセス |
| **データベースのパス** | Mac では `/Users/anna/Dropbox/accounts.obk`、PC では `D:\Dropbox\accounts.obk` |
| **キーファイルのパス** | `/Volumes/USB/key.obkey` または `E:\key.obkey` |
| **独自のバックアップ** | そのデバイスのフォルダ、または共通の設定（第 12 章） |

## 「デバイスとパス」ウィンドウ

データベースや金庫の ⋯ メニューから**デバイスとパス…**を選びます：

- 登録されているすべてのデバイスが、システム、バージョン、最終アクセス、キーファイルの使用、バックアップ先とともに表示されます；
- 目のアイコンで各デバイスのパスを表示し、鉛筆でこのデバイスの名前を変更します；
- ゴミ箱で、もう使わないデバイスを**削除**します。そのパスとバックアップがファイルから取り除かれます（再び開くと改めて登録されます）。




**図 9** データベースを使っているデバイス。

**なぜ必要か**

Dropbox 内の同じデータベースでも、Mac・Windows・Linux ではパスが異なり、キーファイルの入った USB メモリはドライブ文字や名前が変わります。デバイスごとの記録により、ほかのデバイスの記録を上書きせずに、それぞれが自分のパスを見つけられます。

データの安全

# 12 自動バックアップ

各データベースと金庫には、このコンピュータ（外付けディスクなど）または設定済みの任意のサーバーに、**2 つ目の保存先**を設定できます。変更のたびに Obskura がバックグラウンドで新しい版をコピーするので、作業が遅くなることはありません。

1. データベースや金庫の ⋯ メニューから**自動バックアップ…**を選びます。
2. **自動バックアップを有効にする**をオンにして、**このコンピュータ上…**か**サーバー上…**でフォルダを選びます。
3. 保存先がサーバーなら、**この設定をすべてのデバイスで使う**にチェックできます。
4. データベースの場合は、保持する**日付付きコピー**の数を選びます。
5. **保存**を押します。**今すぐ実行**で完全なコピーを強制できます。

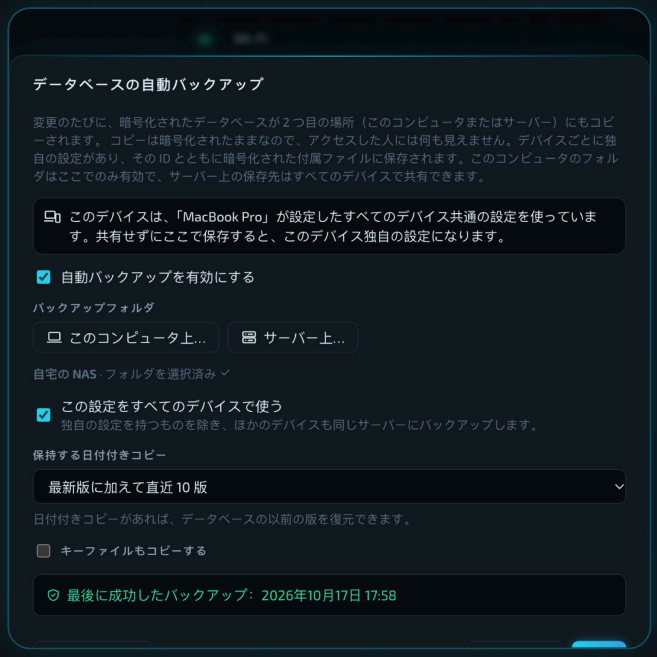



図 10　すべてのデバイスで共通の、NAS へのバックアップ。

| 対象 | コピー方法 |
|---|---|
| **データベース** | バックアップフォルダ内の `.obk` ファイルと、日付付きコピー `名前.YYYYMMDD-hhmmss.obk`。上限を超えた古いものは削除されます。 |
| **金庫** | 暗号化フォルダの複製。変更のたびに、変更のあったファイルだけをコピーし直します。コピーは通常の金庫として開けます。 |
| **キーファイル** | 任意：設定を保存したときに一度だけ、指定したフォルダにコピーします。 |

## デバイスごとのバックアップ

- 各デバイスには**独自の設定**があり、その ID とともに付属ファイルに保存されます。ローカルフォルダ（Mac の外付けディスクなど）は、そのデバイスでのみ有効です。
- サーバー上の保存先は**共通**の設定にできます。独自の設定を持つデバイスを除き、ほかのデバイスもそれを使います。ウィンドウには誰が設定したかが表示されます。
- コピーは**暗号化**されたままなので、バックアップ先を信頼する必要はありません。
- バックアップに失敗すると警告が表示され、Obskura は次の変更時に再試行します。

**キーファイルとバックアップ**

キーファイルをバックアップと同じフォルダに置かないでください。そのフォルダにアクセスできる人が、データベースと認証情報の半分を手にすることになります。

ほかのプログラムとの間で

# 13 読み込みとエクスポート

## オンラインのパスワード管理から読み込む

Obskura は以前のサービスに直接サインインし、暗号化された保管庫をダウンロードして、あなたのコンピュータで復号します。平文でエクスポートしたファイルを経由しません。認証情報はそのサインインにのみ使われ、保存されません。

1. 読み込み先のデータベースを開き、⋯ → **オンラインのパスワード管理から読み込む⋯** を選びます。
2. サービスを選んでサインインします。2 段階認証を使っている場合はコードを求められます。
3. プレビューでエントリ数、フォルダ、例を確認します。
4. フォルダをグループとして再現するか、既存のエントリをスキップするかを決め、**読み込む**を押します。

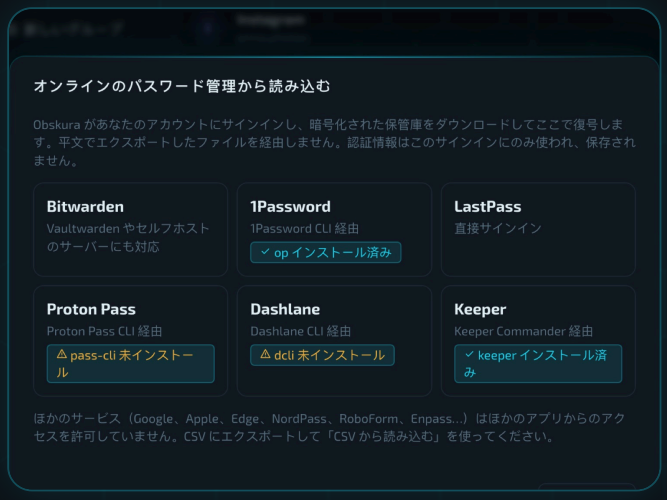



図 11 読み込み元のサービスの選択。

| サービス | アクセス方法 |
|---|---|
| Bitwarden / Vaultwarden | 直接。bitwarden.eu やセルフホストのサーバーにも対応。個人 API キーも使えます。 |
| LastPass | 直接。2 段階認証（コードまたはスマートフォンでの承認）に対応。 |
| 1Password | 1Password CLI（`op`）経由。1Password アプリで許可するか、メールアドレス・パスワード・Secret Key を使います。 |
| Proton Pass | Proton Pass CLI（`pass-cli`）経由。ブラウザでサインインします。 |
| Dashlane | Dashlane CLI（`dcli`）経由。先にターミナルで一度 `dcli sync` を実行します。 |
| Keeper | Keeper Commander（`keeper`）経由。 |

フォルダ、カスタムフィールド、TOTP コード、カード、メモは保持されます。ほかのサービス（Google、Apple、Edge、NordPass⋯）はほかのアプリからのアクセスを許可していないため、CSV にエクスポートしてください。

## CSV から読み込む（KEEPASS など）

1. 元のプログラムで CSV にエクスポートします：KeePass 2 は**ファイル** → **エクスポート** → **KeePass CSV (1.x)**、KeePassXC は**データベース** → **エクスポート** → **CSV ファイル**、Bitwarden、1Password、Chrome、Edge、Firefox はそれぞれの設定から行います。
2. Obskura でデータベースを開き、⋯ → **CSV から読み込む（KeePass⋯）⋯**を選びます。
3. ファイルを選ぶと、Obskura が形式と列を認識してプレビューを表示します。
4. ファイルのグループを再現するか、既存のエントリをスキップするかを決め、**読み込む**を押します。




**図 12** KeePassXC のエクスポートのプレビュー。

グループ、タグ、TOTP コード（保護フィールドとして保存）、作成日、有効期限は保持され、認識できない列はカスタムフィールドになります。同じファイルを再度読み込んでも重複は作られません。

## CSV にエクスポート

⋯ → **CSV にエクスポート⋯**は、KeePassXC の列にタグ、有効期限、お気に入り、カスタムフィールドを加えたファイルを作成します。Obskura、KeePassXC、ほとんどのパスワードマネージャーに再度読み込めます。

**CSV は暗号化されていません**

このファイルにはすべてのパスワードが平文で含まれます。データの移行にだけ使い、すぐに削除してください。読み込んだあとは、古いプログラムからエクスポートした CSV も削除してください。バックアップには、暗号化されたままの自動バックアップを使ってください。

# 14 カスタマイズ 言語

Obskura は 8 つの言語に対応し、さらに追加することもできます。アプリのテキストはすべて、言語ごとに 1 つの XML ファイルに入っています。

## 言語を選ぶ

起動時、Obskura はシステムの言語を使います。含まれていない言語の場合は英語を使います。**設定** → **言語**では**自動**のままにするか、言語を選べます。変更はエラーメッセージも含めてすぐに反映されます。アラビア語は右から左に読むため、画面もそれに合わせて左右反転します。

| | |
|---|---|
| Italiano | it |
| English | en |
| Deutsch | de |
| Español | es |
| Русский | ru |
| 中文（简体） | zh |
| 日本語 | ja |
| العربية | ar |

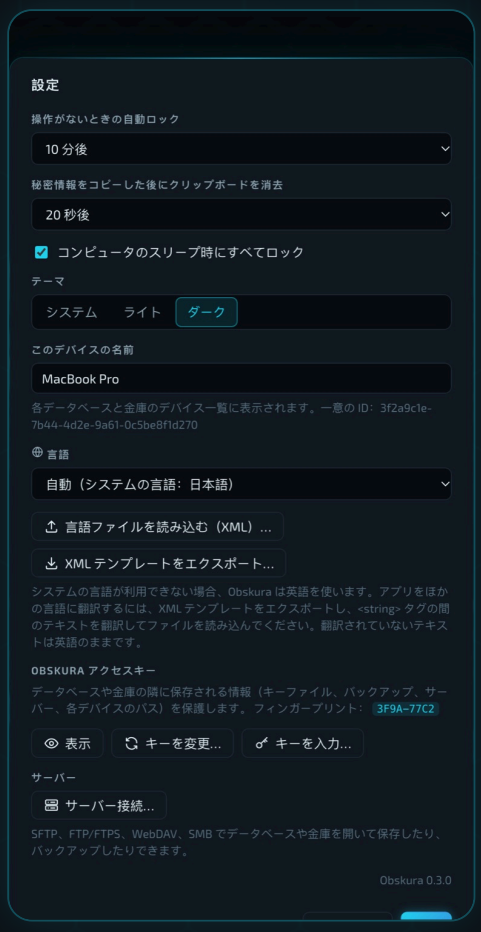



図 13 設定の「言語」セクション。

## 言語を追加する

1. **設定** → **言語**で**XML テンプレートをエクスポート…**を押すと、現在の言語の完全なファイルが得られます（または、すべての言語の XML を収めた zip をウェブサイトからダウンロードします）。
2. ヘッダーの `code`（言語コード。例：`fr`、`pt-BR`）と `name` を変更し、右から左に書く言語では `dir="rtl"` を設定します。
3. 任意のテキストエディタで各 `<string>` 要素のテキストを翻訳し、UTF-8 で保存します。
4. **言語ファイルを読み込む（XML）…**を押すと、その言語がすぐに一覧に表示され、その言語のシステムでは自動で使われます。

## ファイルの形式

言語ファイルは、`<language>` 要素と、テキストごとの `<string>` から成る XML です：

```
<?xml version="1.0" encoding="UTF-8"?>
<language code="fr" name="Français" english-name="French" dir="ltr">
  <string key="common.cancel">Annuler</string>
  <string key="db.syncFailed">Échec de la synchronisation : {message}</string>
  <string key="vault.itemsN.one">{count} élément</string>
  <string key="vault.itemsN.other">{count} éléments</string>
</language>
```

- `key` 属性は翻訳も変更もしないでください；
- `{count}` や `{name}` のような波かっこ内のプレースホルダーはそのまま残してください。アプリが数値や名前に置き換えます；
- テキスト内の `<b>` と `<code>` タグは残してください；
- `.one`、`.two`、`.few`、`.many`、`.other` で終わるキーは複数形です。ご自身の言語に必要な形を使い、`.other` は必須です；
- 足りないテキストは英語で表示されるので、一部だけの翻訳でもそのまま使えます。

## 内蔵の言語を修正する

内蔵の言語と同じ `code`（例：`ja`）の XML ファイルは、その言語を置き換えず、含まれている項目だけを修正します。設定には**修正**と表示され、× で削除すると元のテキストに戻ります。

> **ヒント**
>
> 読み込んだファイルはアプリのデータフォルダに保存され、そのコンピュータでのみ有効です。ほかのコンピュータでも読み込んでください。

## 空の XML テンプレートで翻訳する

新しい言語は、ウェブサイト（`obskura.bytenooks.com`、「ダウンロード」セクション）からダウンロードできる空のテンプレート `obskura-translation-template.xml` から始められます。アプリのテキストごとに 1 行あり、訳文は `<string …>` と `</string>` の間に書きます。属性 `en` は英語の参照テキストで、そのまま残してかまいません（アプリは無視します）。

```
<language code="fr" name="Français" english-name="French" dir="ltr">
  <string key="common.lock" en="Lock">Verrouiller</string>
  <string key="common.settings" en="Settings">Paramètres</string>
  <string key="common.cancel" en="Cancel">Annuler</string>
</language>
```

1. テンプレートをダウンロードし、言語コードの名前で保存します（例：`fr.xml`）。
2. 最初の `<language>` 行で `code`、`name`、`english-name` を入力し、必要なら `dir="rtl"` を指定します。
3. UTF-8 で保存できるテキストエディターで、上記のルールに従って訳文を書きます。
4. **翻訳しない行は削除してください**。空のテキストはアプリで空白として表示されますが、行がなければ英語で表示されます。
5. **設定** → **言語**でファイルを読み込み、アプリでテキストを確認します。

**翻訳を共有する**

ウェブサイトのお問い合わせページから開発者に送ってください。トピックで「翻訳」を選び、XML ファイルを添付します（最大 3 MB。完全な言語ファイルは約 100〜150 KB）。届いた翻訳は今後のバージョンに含まれることがあります。

セキュリティ

# 15 ロックと設定

データベースや金庫をロックすると、Obskura はセッションを閉じ、鍵と復号したデータを**メモリから消去**します。再び開くには、もう一度認証情報が必要です。

## ロックの方法

- サイドバー下部の**すべてロック**、または `Ctrl/⌘` + `L`。
- サイドバーでロック解除中の項目の横にある錠前、または ⋯ メニューの**ロック**。
- 一定時間操作がないとき、またはコンピュータがスリープしたときに**自動で**。

## 設定

| 設定 | 効果 | 既定値 |
|---|---|---|
| **操作がないときの自動ロック** | マウスやキーボードを使わずに 1〜60 分経つとすべてロック、または無効。 | 10 分 |
| **クリップボードを消去** | 秘密情報をコピーして 10〜120 秒後にクリップボードを消去、または無効。 | 20 秒 |
| **スリープ時のロック** | コンピュータがスリープするとすべてロック。 | オン |
| **テーマ** | システム、ライト、ダーク。 | ダーク |
| **このデバイスの名前** | 各データベースと金庫のデバイス一覧に表示され、その下に一意の ID が表示されます。 | コンピュータ名 |
| **言語** | 自動、または利用可能な言語のいずれか。XML ファイルの読み込みとエクスポート（第 14 章）。 | 自動 |
| **Obskura アクセスキー** | キーの表示、変更、入力（第 11 章）。 | — |
| **サーバー** | SFTP、FTP/FTPS、WebDAV、SMB の接続（第 09 章）。 | — |

設定には秘密情報は含まれません。サーバーのパスワードはシステムのキーチェーンにあり、最近使った項目の一覧はアクセスキーで暗号化されています。

リファレンス

# 16 ショートカットと暗号

## キーボードショートカット

| キー | 場所 | 動作 |
|---|---|---|
| `Ctrl/⌘` + `L` | どこでも | すべてロック |
| `Esc` | ダイアログ | ウィンドウを閉じる |
| `Ctrl/⌘` + `F` | データベース | 検索 |
| `Ctrl/⌘` + `N` | データベース | 新しいエントリ |
| `Ctrl/⌘` + `S` | エントリエディタ | エントリを保存 |
| ダブルクリック | エントリー覧 | パスワードをコピー |
| `Ctrl/⌘` + `A` | 金庫 | すべて選択 |
| `Enter` | 金庫 | ファイルまたはフォルダを開く |
| `F2` | 金庫 | 名前を変更 |
| `Del` · `⌘` + `⌫` | 金庫 | 選択した項目を削除 |
| `⌫` | 金庫 | 上のフォルダへ |

## 暗号の詳細

| 構成要素 | アルゴリズム |
|---|---|
| **鍵導出** | Argon2id：256 MiB / 3 パス、1 GiB / 4 パス、2 GiB / 6 パス、256 ビットのランダムなソルト。 |
| **複合鍵** | パスワード（NFC 正規化）とキーファイルの SHA-256。KeePass KDBX4 と同じ方式。 |
| **暗号化** | AES-256、XChaCha20、Serpent、Twofish、Camellia、またはカスケード。常に認証付き（GCM または Poly1305）。カスケードでは各層が HKDF で導出した独自の鍵を持ちます。 |
| **サブ鍵** | HKDF-SHA256。保存のたびに新しいシードを使用。 |
| **完全性** | 認証付きヘッダー：少しでも改ざんされると開けません。 |
| **金庫内の名前** | AES-SIV。格納フォルダに結び付けて暗号化。 |
| **ファイルの内容** | 64 KiB 単位のブロック、ファイルごとのランダムな鍵、並べ替えと切り詰めへの保護。 |
| **付属ファイル** | Obskura アクセスキー（256 ビットの乱数）から HKDF で導出した鍵による AES-256-GCM。 |
| **メモリ** | 鍵と平文のデータはネイティブプロセス内にのみ存在し、終了時に消去されます。 |

サポート

# 17 推奨事項とトラブルシューティング

## 推奨事項

- **自動バックアップを有効にする**か、`.obk` ファイルと金庫のフォルダを定期的に外付けディスクにコピーしてください。すでに暗号化されているので、どこに保管しても安全です。
- **キーファイルはデータベースと別に保管し**、予備のコピーを少なくとも 1 つ用意してください。
- **Obskura アクセスキーを保管する**：紙に書くか、コンピュータ以外の場所に保存してください。
- **サービスごとに異なるパスワードを使う**：生成ツールで作れば、覚えるのは 1 つだけで済みます。
- **席を離れるときはロックする**：`Ctrl/⌘` + `L` が最も手早い方法です。
- もう使わないデバイスは、**デバイスとパス**ウィンドウから**削除**してください。

## よくある質問

**マスターパスワードを忘れました。復元できますか？**

できません。それこそが Obskura を安全にしている点です。正しい認証情報がなければ、開発者を含め誰もデータベースや金庫を開けません。

**「パスワードまたはキーファイルが違います」と表示されるが、正しいはず。**

作成時と同じキーファイルを選んでいるか、変更されていないかを確認してください。パスワードとキーファイルを使った場合は両方が必要です。Caps Lock も確認してください。

**開くのに数秒かかります。正常ですか？**

はい。これは Argon2id の処理時間で、パスワードの推測を遅らせるための意図的なものです。「高」レベルでは数秒と 1 GiB の空きメモリ、「最高」では最大 12 秒と 2 GiB が必要です。

**別のコンピュータでキーファイルが見つかりません。**

キーファイルの入ったドライブを接続してください。Obskura は、そのコンピュータが前回使ったパス、データベースの隣、ほかのデバイスのパス、バックアップコピーの中を探します。それでも見つからない場合は**選択…**で選べば、次回からそのデバイス用にパスが記憶されます。

#### 「別のアクセスキーで保護されています」と表示されます。

別のコンピュータでアクセスキーが変更されています。**現在のキーを入力**をクリックして新しいキーを使ってください。パスワードでロックを解除し、キーファイルを手動で選ぶこともできます。

#### 「別の場所で認証情報が変更されました」と表示されました。

誰かが別のデバイスでデータベースのパスワードまたはキーファイルを変更しました。**ロック**を選び、新しい認証情報で開き直してください。**上書き**を選ぶと、別の場所での変更が取り消されます。

#### サーバーに接続できません。

**接続テスト**でアドレス、ポート、認証情報を確認してください。WebDAV ではベースパス（Nextcloud は `/remote.php/dav/files/<ユーザー>`）、SMB では共有名を確認します。自己署名証明書の場合、**自己署名証明書を受け入れる**は信頼できるネットワークでだけ有効にしてください。

#### 金庫内のファイルへの変更が保存されていないようです。

もう一方のアプリでファイルを保存したことを確認し、**変更を金庫に保存**を押してください。ファイルを開いたまま金庫をロックした場合でも、Obskura は一時コピーを削除する前に保存を試みます。

#### アプリが自分の言語で起動しません。

Obskura はシステムの言語に従い、含まれていない言語の場合は英語を使います。**設定** → **言語**で言語を選ぶか、XML ファイルを読み込んでください（第 14 章）。

#### MACOS や WINDOWS で、アプリが検証されていないと表示されます。

アプリは商用の証明書で署名されていません。初回は第 02 章の手順に従って開いてください。



# OBSKURA

あなたのデータに見えないセキュリティを

パスワードマネージャーと暗号化ファイル金庫 · あなたのコンピュータやサーバーで

macOS · Windows · Linux · バージョン **0.3.0**

https://obskura.bytenooks.com