CLOUDFLARE 連続勉強会

Cloudflare Zero Trust の紹介

前編（導入準備編）＋ 後編（導入編）

Zero Trust 導入フロー / WARP / ZTNA / SWG / CASB / DLP ― ダッシュボード画面・アーキテクチャ図付きで解説

Zero Trust | WARP | ZTNA | SWG | CASB | DLP | Logpush | Enterprise

⚠ **対象：**Zero Trust 入門者 ｜ **前提：**基本的なネットワーク知識

※ 2024年 Webinar ベース。最新情報は Docs を参照

目次 ― 全10セッション

前編

前編

Cloudflare Zero Trust の紹介（前編）

導入準備編 ― 導入フロー / ダッシュボード設定 / WARP クライアント

#6

Cloudflare Zero Trust の紹介（前編）

導入準備編 ― 導入フロー / ダッシュボード設定 / WARP クライアント

#6

1 製品名変更のお知らせ

⚠ **【2026年2月18日より製品名が変更されました】**

Cloudflare One 製品スイートの名称が以下の通り変更されています：

- **WARP Client → Cloudflare One Client**
- **WARP Connector → Cloudflare Mesh**
- **Magic WAN → Cloudflare WAN / Cloudflare IPsec / Cloudflare GRE**
- **Magic WAN Connector → Cloudflare One Appliance**
- **Magic Firewall → Cloudflare Network Firewall**

※ 本資料では旧名称（WARP 等）を使用している箇所があります。
お客様側での対応は不要です。既存の設定、機能、課金はすべてそのまま維持されます。API および Terraform のリソース名も変更ありません。
Cloudflare Docs の詳細はこちら ↗

2 Zero Trust ポートフォリオ

1 製品名変更のお知らせ

⚠ **【2026年2月18日より製品名が変更されました】**

Cloudflare One 製品スイートの名称が以下の通り変更されています：

- **WARP Client → Cloudflare One Client**
- **WARP Connector → Cloudflare Mesh**
- **Magic WAN → Cloudflare WAN / Cloudflare IPsec / Cloudflare GRE**
- **Magic WAN Connector → Cloudflare One Appliance**
- **Magic Firewall → Cloudflare Network Firewall**

※ 本資料では旧名称（WARP 等）を使用している箇所があります。
お客様側での対応は不要です。既存の設定、機能、課金はすべてそのまま維持されます。API および Terraform のリソース名も変更ありません。
Cloudflare Docs の詳細はこちら ↗

2 Zero Trust ポートフォリオ

誰も信頼しない
Zero Trust 原則

グローバルエッジ
統合セキュリティ

どこからでも安全
場所を問わないアクセス

2 Zero Trust ポートフォリオ

🚫
誰も信頼しない
Zero Trust 原則

🌍
グローバルエッジ
統合セキュリティ

🔒
どこからでも安全
場所を問わないアクセス

コアコンポーネント

🔐 ZTNA（Cloudflare Access）

Zero Trust Network Access ― VPN を使わずに、アプリケーションへのセキュアなアクセスを提供します。

- アプリケーション単位のアクセス制御
- IdP 連携による認証・認可
- デバイスポスチャーの検証
- セッションごとの再認証

対象: Self-hosted / SaaS / Private Network アプリ

🛡️ SWG（Cloudflare Gateway）

Secure Web Gateway ― インターネットへのアウトバウンドトラフィックを検査・制御します。

- DNS / HTTP / Network フィルタリング
- マルウェア・フィッシング対策
- AV スキャン・サンドボックス
- TLS 復号化による可視化

対象: 全てのインターネットトラフィック

📱 WARP Client（Cloudflare One Client）

エンドポイントエージェント ― デバイスを Cloudflare ネットワークに接続するクライアントソフトウェアです。

- WireGuard / MASQUE トンネル
- DNS over HTTPS（DoH）
- Device Posture 情報の収集
- Split Tunnel / Local Domain Fallback

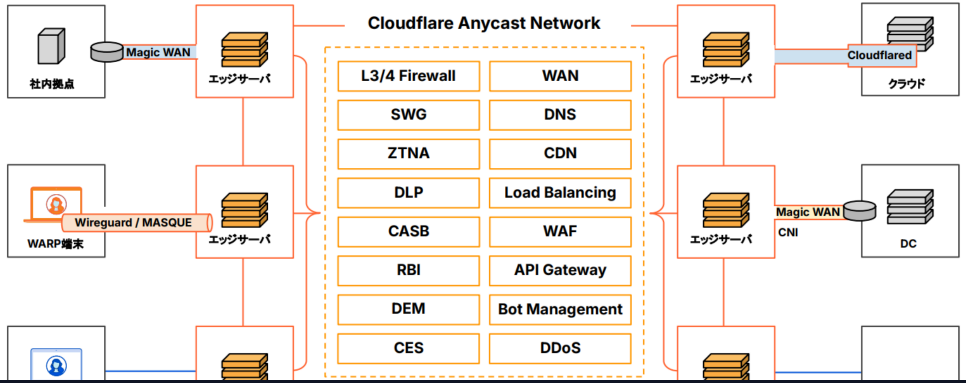
対応 OS: Windows / macOS / Linux / iOS / Android

データ保護・SaaS セキュリティ

3 Cloudflare Zero Trust 導入の流れ

導入の流れとしては 📖 Roadmap to Zero Trust をご参照ください。Zero Trustへのロードマップは、複数のフェーズにわたるステップで構成されています。Zero Trust は「ユーザー・デバイス・コンテキスト」の 3 要素を検証し、アプリケーションへのアクセスを制御します。

Cloudflare One - 包括的なSASEソリューション

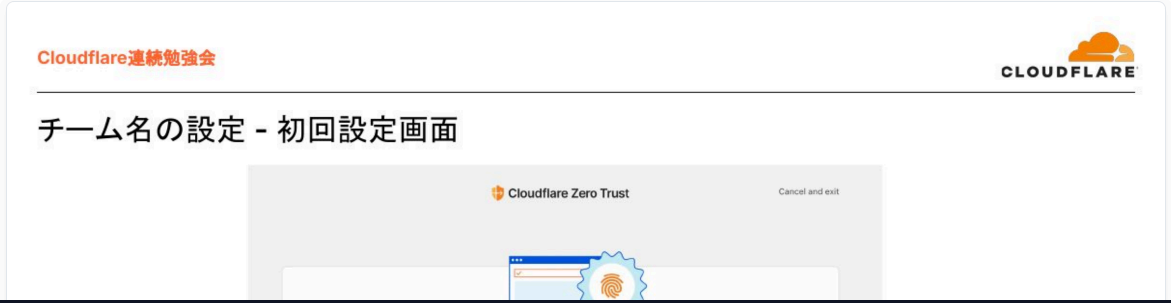


3 Cloudflare ダッシュボードの初期設定

チーム名の設定

Cloudflare ダッシュボードのアカウントメニューから **Zero Trust** を選択し、Team 名を決定します。Team 名は後から変更可能ですが、`yourteam.cloudflareaccess.com` の URL に使用されます。

1. Cloudflare ダッシュボードから **Zero Trust** をクリック
2. Team 名を指定
3. サブスクリプションプランとお支払い方法を設定



Gateway DNS	Gateway で検出された DNS クエリ情報
Gateway HTTP	Gateway で検出された HTTP リクエスト
Gateway Network	Gateway で検出されたネットワークセッション情報
SSH Logs	Access for Infrastructure 経由の SSH セッションログ
WARP Config Changes	WARP クライアントのデバイスプロファイル設定変更
WARP Toggle Events	WARP クライアントの有効化/無効化イベント
Zero Trust Network Session Logs	Zero Trust ネットワークセッションの詳細ログ

📖 Cloudflare Docs — Logpush integration

重要なログフィールド

ログタイプ	フィールド	説明
Gateway DNS / Gateway HTTP	`Action`	Allow / Block / Isolate 等のアクション
Gateway DNS / Gateway HTTP	`UserEmail`	リクエストを行ったユーザー
Gateway DNS / Gateway HTTP	`DeviceID`	リクエスト元デバイスの識別子
Gateway DNS / Gateway HTTP	`PolicyName`	適用されたポリシー名
Access	`Allowed`	アクセスが許可されたかどうか
Access	`AppUID`	アクセス先アプリケーションの ID

ユーザー名

ユースケース：

- 特定のデバイスで接続問題が発生した際の状態確認
- 古い WARP バージョンを使用しているデバイスの特定
- 退職者のデバイスが残っていないかの監査

Cloudflare Docs — Device analytics

数

ユースケース：

- ライセンス（シート）使用状況の確認
- 長期間ログインしていないユーザーの特定
- 不正アクセスの兆候がないかの監視

Cloudflare Docs — User analytics

WARP CGNAT IP 範囲

WARP クライアントが接続すると、デバイスには **CGNAT（Carrier-Grade NAT）** 範囲の仮想 IP アドレスが割り当てられます。この IP アドレスは、Cloudflare のネットワーク内でデバイスを一意に識別するために使用されます。

Cloudflare Docs — WARP client IP addresses

プロトコル	CGNAT IP 範囲	用途
IPv4	`100.96.0.0/12`	WARP 接続デバイスの仮想 IPv4 アドレス
IPv6	`fd00::/8`（ULA 範囲）	WARP 接続デバイスの仮想 IPv6 アドレス

CGNAT の役割：

- **デバイス識別** — 各 WARP 接続デバイスに一意の仮想 IP を割り当て、ポリシー適用やログ記録に使用
- **プライベートネットワークアクセス** — Tunnel 経由でプライベートリソースにアクセスする際の送信元 IP として機能
- **Gateway ポリシー** — ユーザー/デバイス単位でのトラフィック制御が可能

重要：CGNAT IP 範囲（ `100.96.0.0/12` ）は RFC 6598 で定義されたキャリアグレード NAT 用の共有アドレス空間です。この範囲は Split Tunnel の Exclude リストに追加しないでください。追加すると WARP 接続が正常に動作しなくなります。

Zero Trust ダッシュボードから、登録されたデバイスやユーザーの状態をリアルタイムで確認できます。

登録デバイス

Team & Resources > Devices

確認できる情報：WARP バージョン、最終ログイン日時、利用ユーザー、OS 情報、デバイス名

ユースケース：

- 特定のデバイスで接続問題が発生した際の状態確認
- 古い WARP バージョンを使用しているデバイスの特定
- 退職者のデバイスが残っていないかの監査

📖 Cloudflare Docs — Device analytics

登録ユーザー

Team & Resources > Users

確認できる情報：アクティブユーザー、最終ログイン日時、ロケーション、紐付けデバイス数

ユースケース：

- ライセンス（シート）使用状況の確認
- 長期間ログインしていないユーザーの特定
- 不正アクセスの兆候がないかの監視

📖 Cloudflare Docs — User analytics

WARP CGNAT IP 範囲

WARP クライアントが接続すると、デバイスには **CGNAT（Carrier-Grade NAT）** 範囲の仮想 IP アドレスが割り当てられます。この IP アドレスは、Cloudflare のネットワーク内でデバイスを一意に識別するために使用されます。

📖 Cloudflare Docs — WARP client IP addresses

プロトコル	CGNAT IP 範囲	用途
IPv4	`100.96.0.0/12`	WARP 接続デバイスの仮想 IPv4 アドレス
IPv6	`fd00::/8`（ULA 範囲）	WARP 接続デバイスの仮想 IPv6 アドレス

いつ使うか： Tunnel の接続問題、パフォーマンス問題、または予期しない動作をトラブルシューティングする際に使用します。通常のログでは情報が不足している場合、デバッグレベルのログを有効化することで、詳細な接続情報、エラーの原因、リクエストの処理状況などを確認できます。

✅ **デバッグログで確認できる情報：** Tunnel の接続状態、Cloudflare エッジとの通信詳細、リクエストのルーティング、エラーの詳細原因など。サポートへの問い合わせ時にこのログを添付することで、問題の特定が迅速になります。

```
# Linux — --loglevel debug を追記して再起動
# cloudflared.service ファイルを編集し、ExecStart 行に --loglevel debug を追加
$ vi /etc/systemd/system/cloudflared.service
# 例: ExecStart=/usr/bin/cloudflared --loglevel debug tunnel run ...
$ systemctl daemon-reload
$ systemctl restart cloudflared.service

# ログ取得（cloudflared がインストールされた端末から）
$ cloudflared tunnel login
$ cloudflared tail <UUID>
$ cloudflared tail <UUID> > tunnel_log.txt
```

⚠️ **注意：** デバッグログは大量の情報を出力するため、本番環境での長期間の有効化は推奨しません。問題解決後は通常のログレベルに戻してください。

📖 Cloudflare Docs — Tunnel Log

接続の問題がある場合：ファイアウォール設定の確認

Cloudflare Tunnel は `cloudflared` からの送信（Egress）トラフィックのみを許可し、受信（Ingress）トラフィックをすべてブロックする「ポジティブセキュリティモデル」を実装できます。接続に問題がある場合は、以下の IP/ポートがファイアウォールで許可されているか確認してください。

1. **DNS ポリシー**（解決前に評価されるセレクター）
2. **Resolver ポリシー**（該当する場合・Enterprise のみ）
3. **DNS ポリシー**（解決後に評価されるセレクター）
4. **Egress ポリシー**（該当する場合・Enterprise のみ）
5. **Network ポリシー**
6. **HTTP ポリシー**

> ⚠️ DNS / Resolver ポリシーは独立して動作します。DNS ポリシーでサイトをブロックしても、対応する HTTP ポリシーがなければ、IP アドレスを知っているユーザーはサイトにアクセスできます。

HTTP ポリシー内の適用順序

HTTP ポリシーは、アクションタイプと優先順位の組み合わせで適用されます：

1. **Do Not Inspect** ― 最初に評価。マッチした場合、復号化をバイパスし他の HTTP ポリシーをスキップ
2. **Isolate** ― リモートブラウザ分離にルーティング（Browser Isolation アドオン）
3. **Allow / Block / Do Not Scan** ― 分離・非分離トラフィック両方に適用
4. **DLP / AV スキャン / サンドボックス** ― HTTP リクエストのボディを検査

優先順位（Precedence）の原則

- **トラブルシューティング時** — アプリが TLS 検査で動作しない場合、まず Do Not Inspect で問題が解決するか確認。ログが残るため原因特定が容易
- **可視性の維持** — 検査はできなくても、どのユーザーがいつどのサービスにアクセスしたかを把握したい場合
- **段階的な除外** — Split Tunnel で完全除外する前に、Do Not Inspect でログを確認しながら影響を評価
- **コンプライアンス要件** — 特定のトラフィックを検査できないが、アクセス記録は必要な場合

✅ **推奨：**問題が発生したアプリは、まず Do Not Inspect で対応し、ログを確認しながら原因を特定してください。Split Tunnel は最終手段として使用することで、セキュリティの可視性を維持できます。

4 Remote Browser Isolation（RBI）

RBI の概要

Remote Browser Isolation（RBI）は、Web ブラウジングをクラウド上の隔離環境で実行し、エンドポイントを保護するセキュリティ技術です。ユーザーのブラウザには安全なレンダリング結果のみが送信されるため、マルウェアや悪意のあるコードがローカルデバイスに到達することを防ぎます。

RBI を適用するには、**Gateway > Firewall Policies > HTTP** でポリシーを作成し、アクションとして **Isolate** を選択します。特定のカテゴリ（例：セキュリティリスク、未分類サイト）や、特定のドメイン・URL に対して Isolate アクションを適用することで、リスクの高い Web コンテンツを隔離環境で安全に閲覧できます。

ℹ️ **プラン制限：**Browser Isolation は **Pay-as-you-go** および **Enterprise プラン**のアドオンとして利用可能です。Free プランでは利用できません。

モード	動作方式	特徴
API モード	SaaS アプリの API に直接接続してスキャン	エージェント不要、既存データも検査可能、定期的な自動スキャン
インラインモード	Gateway 経由のトラフィックをリアルタイム検査	リアルタイム保護、DLP と連携、アップロード/ダウンロード時に検査

CASB で検出できるリスク

⚠️ **設定ミス**
公開共有リンク、外部ユーザーへの過剰なアクセス権限、MFA 未設定のアカウント

👻 **シャドー IT**
未承認の SaaS アプリの利用状況を検出

🔓 **データ漏洩リスク**
機密ファイルの外部共有、個人アカウントへのデータ転送

📋 **コンプライアンス違反**
データ保持ポリシー違反、地域制限違反

対応 SaaS アプリケーション（例）

Microsoft 365 Google Workspace Salesforce Box Slack GitHub Dropbox Zoom OpenAI ChatGPT Anthropic Claude Google Gemini Bitbucket Cloud AWS S3

Findings の深刻度レベル（Severity）

深刻度	内容	対応の優先度

生成 AI へのファイルアップロード時に内容をスキャン

どのユーザーが何を入力しようとしたかを記録（監査用）

⚠ **注意：**生成 AI サービスへの DLP 適用には、HTTP ポリシーで対象ドメイン（`chat.openai.com`、`claude.ai` 等）を指定し、DLP Profile を適用する必要があります。

🎬 関連デモ動画

- YouTube **AI データ保護の概要（DLP）** ― PII やソースコードなどの機密データ漏洩をブロックし、Guardrails でリスクのあるプロンプトを検出
- YouTube **デモ：DLP 設定ウォークスルー（DLP）** ― 生成 AI アプリ利用時の機密データ保護設定を実演
- YouTube **デモ：シャドー AI の検出（DLP + CASB）** ― 未承認 AI ツールの利用状況を可視化し、制御を取り戻す方法
- YouTube **生成 AI アプリのセキュリティ態勢管理（CASB）** ― ChatGPT、Claude、Gemini の設定ミスやデータ漏洩リスクを検出
- YouTube **デモ：生成 AI の CASB 連携設定（CASB）** ― API 連携による生成 AI ツールのセキュリティ態勢管理を実演

DLP の導入方式

方式	説明	要件
インライン検査	Gateway HTTP ポリシーでリアルタイムにトラフィックをスキャン	WARP クライアント + TLS 検査の有効化
CASB 連携	SaaS アプリ内の既存データを API 経由でスキャン	CASB Integration の設定

✅ **導入のポイント：**まず Predefined Profile（クレジットカード、社会保障番号など）を有効化し、ログモードで運用を開始してください。誤検知の状況を確認しながら、段階的にブロックモードへ移行することを推奨します。

Cloudflare Docs ― Data Loss Prevention

Cloudflare Docs — Data Loss Prevention

DLP Profile の設定

1. DLP > DLP Profile から対象の **Managed Profile** を選択 → 「Configure」
2. 有効化したい Detection Entry を選択 → 「Save Profile」
3. 必要に応じて Custom Profile を追加

よく使われる Managed Profile

プロファイル名	検出対象	ユースケース
Financial Information	クレジットカード番号、銀行口座番号	PCI-DSS コンプライアンス対応
Credentials and Secrets	API キー（AWS / GCP / Azure）、SSH キー、パスワード	クラウド認証情報の漏洩防止
Source Code	Python、JavaScript、Java、Go、Rust 等のコード	知的財産の保護、生成 AI への入力防止
AI Prompt	Security、Customer、Financial、PII、Technical 情報	生成 AI への機密情報入力を検出

一般的な DLP ポリシー例

ポリシー	説明	アクション

Cloudflare Docs — Create a DLP policy

DLP ログの確認

1. Logs > Gateway > HTTP で DLP ポリシーを選択して検索
2. 該当レコードを選択 → 「Decrypt payload log」で Private Key を入力

CASB との連携 — Scan for sensitive data

CASB と DLP を組み合わせることで SaaS アプリ上の機密データ漏洩を検知できます。CASB Findings および DLP ログの双方で確認可能です。

Cloudflare Docs — Scan for sensitive data

Automation（自動化）

デプロイと設定変更、あるいはロールバックを自動化するには、以下を利用します：

Cloudflare API

すべての設定をプログラムから管理するための RESTful API。

Terraform

Infrastructure as Code での管理。Dashboard UI から Terraform への移行には cf-

WARP UI	`onboarding`	初回起動時のプライバシーポリシー確認画面の表示/非表示	`false`（非表示）
WARP 接続	`environment`	FedRAMP High 環境への接続設定	`fedramp_high`
WARP 接続	`override_api_endpoint`	API 通信先 IP のオーバーライド（中国パートナー向け等）	`1.2.3.4`
WARP 接続	`override_warp_endpoint`	WARP トラフィック送信先のオーバーライド	`203.0.113.0:500`
DNS	`override_doh_endpoint`	DoH 通信先のオーバーライド（DNS only モード）	`1.2.3.4`
WARP 接続	`enable_pmtud`	Path MTU Discovery の有効化	`true`
Windows	`enable_netbt`	NetBIOS over TCP/IP の有効化（Windows）	`true`
セキュリティ	`enable_post_quantum`	ポスト量子暗号の有効化	`true`
Windows	`multi_user`	Windows での複数ユーザー登録の有効化	`true`
Windows	`pre_login`	Windows ログイン前の WARP 接続	`true`
モバイル	`unique_client_id`	デバイス UUID の割り当て（iOS/Android）	`496c6124-...`
組織管理	`configs`	複数 Zero Trust 組織間の切り替え設定	配列形式

📖 Cloudflare Docs — MDM Parameters

Android Per-app VPN（MDM 専用）

Android デバイスで特定のアプリのみ WARP トンネル経由にする設定は MDM でのみ可能です。

WARP モバイル	`is_browser`	アプリがブラウザかどうかを指定（再認証やブロック通知用）

✅ **ヒント：**MDM パラメータは Dashboard 設定より優先されます。サービストークンによる自動登録（ `auth_client_id` / `auth_client_secret` ）も MDM で設定可能です。

📖 Cloudflare Docs — MDM Parameters

🔧 API 専用機能

以下の機能は Dashboard では設定できず、**API または Terraform でのみ**設定可能です。

API 専用の設定項目

カテゴリ	機能	説明	API エンドポイント	リンク
WARP / mTLS	**Posture Only Mode クライアント証明書**	Posture only モードを有効にするには、API でクライアント証明書のプロビジョニングを有効化する必要がある	`PATCH /zones/{zone_id}/devices/policy/certificates`	Docs ↗
Access	**Legacy Policy 変換**	レガシーポリシーを再利用可能なポリシー（Reusable Policy）に変換	`PUT /accounts/{account_id}/access/apps/{app_id}/policies/{policy_id}/make_reusable`	Docs ↗

カテゴリ	機能	説明	API エンドポイント	リンク
WARP / mTLS	Posture Only Mode クライアント証明書	Posture only モードを有効にするには、API でクライアント証明書のプロビジョニングを有効化する必要がある	`PATCH /zones/{zone_id}/devices/policy/certificates`	Docs ↗
Access	Legacy Policy 変換	レガシーポリシーを再利用可能なポリシー（Reusable Policy）に変換	`PUT /accounts/{account_id}/access/apps/{app_id}/policies/{policy_id}/make_reusable`	Docs ↗
WARP	Custom Device Posture	外部 API を呼び出すカスタムデバイスポスチャチェック（WARP service-to-service integration）	`POST /accounts/{account_id}/devices/posture/integration`	Docs ↗
Gateway	Connectivity Settings	`icmp_proxy_enabled`, `offramp_warp_enabled` の設定	`PATCH /accounts/{account_id}/zerotrust/connectivity_settings`	API ↗
Access	Infrastructure Targets 一括操作	インフラストラクチャーゲットの一括追加・削除	`PUT /accounts/{account_id}/infrastructure/targets/batch`	API ↗
Access	Service Token Rotation	サービストークンの Client Secret をローテーション	`POST /accounts/{account_id}/access/service_tokens/{uuid}/rotate`	API ↗

Dashboard で設定可能だが API でより詳細な制御が可能な機能

カテゴリ	機能	API での追加機能	API エンドポイント	リンク

HTTP/2 を部分的にサポートするオリジンと通信する際に発生することがあります。

原因	説明
HTTP/2 ダウングレード要求	オリジンが HTTP/2 で接続を開始後、一部リクエストで HTTP/1.1 へのダウングレードを要求
IIS サーバー	Microsoft IIS は HTTP/2 経由の認証をサポートしていない場合がある

✅ **解決方法:** オリジンサーバーで HTTP/2 を無効化してください。Gateway は HTTP/1.1 へのダウングレードをサポートしていません。

証明書の警告: 信頼されていない証明書

すべてのページで証明書の警告が表示され、インターネットを閲覧できない場合：

1. Cloudflare ルート証明書がデバイスにインストールされているか確認
2. Team & Resources > Devices > WARP Client で「Install CA to system certificate store」が有効か確認
3. ブラウザを再起動（Chrome/Edge は起動時に証明書をキャッシュ）

ℹ️ HTTPS トラフィック検査には、ユーザーのデバイスに Cloudflare ルート証明書をインストールして信頼する必要があります。

モバイルアプリで証明書エラー

システムに Cloudflare 証明書をインストールしても、一部のモバイルアプリで無効な証明書の警告が表示される場合があります。

SSO	Single Sign-On。複数のアプリケーションログインを1つに統合し、ユーザーは一度だけ資格情報を入力。
SCIM	System for Cross-domain Identity Management。IdP（Okta や Microsoft Entra ID など）がユーザー ID 情報をクラウドアプリと同期できるオープン標準プロトコル。
Service Token	Cloudflare Access が生成する認証資格情報。自動化システムが保護されたアプリケーションにアクセスできるようにする。シートを消費しない。
Access Token	ユーザーに特定の Access アプリケーションへのアクセスを一定期間付与するデータ。ブラウザ Cookie に保存またはパスワードの代わりにアプリに渡される。

デバイス・ポスチャ

用語	説明
Device Posture	ユーザーのデバイスのセキュリティを評価する方法。シリアル番号の検証や最新のソフトウェアアップデートの確認など。
Device Profile	組織内の特定のデバイスセットに適用される WARP クライアント設定のコレクション。
Device Registration	物理デバイス上の WARP クライアントの個別セッション。固有の公開鍵、デバイスプロファイル、仮想 IP アドレス（IPv4 と IPv6 各1つ）を含む。
Fleet	ユーザーデバイスのコレクション。フリート内のすべてのデバイスには WARP がインストールされ、Zero Trust 組織に接続。
MDM	Mobile Device Management。組織がデバイスにインストールされるソフトウェア・設定・証明書を管理できる構成ファイル。

ネットワーク・DNS

用語	説明