ゼロトラストネットワークアクセスの導入を容易にするクラウド型プロキシの開発

― Revolutionary Zero Trust Service ―

1. 背景

近年、新型コロナウイルス感染症の蔓延により、我が国においても情報通信技術を活用したテレワークを導入する企業が増加した。こうした情勢の中、認知されているサイバー攻撃の件数が急激に増加していることが明らかとなっており、警察庁は VPN 機器からの侵入が 71%にのぼると報告する。テレワークの普及により、社外から社内ネットワークへの接続が常態化することで、ネットワーク内外の境界が曖昧となり、脆弱な境界を狙った攻撃が拡大したと推察される。こうした課題に対して有効とされているのが、境界内外問わずアクセスする人間を厳格に検証するゼロトラストネットワークアクセス（Zero Trust Network Access, ZTNA）を含むゼロトラストセキュリティである。このセキュリティモデルは、各情報システムやサービスごとに適切かつ最小権限によるアクセス制御を行うことで、内部脅威を軽減するとともに、BYOD（Bring Your Own Device）を想定した柔軟なアクセス管理を行うなど、多くの利点がある。

一方で、民間企業がゼロトラストを導入する際には時間・人材・予算のコスト不足による問題が生じるとされている。特に中小企業や地方自治体など、予算も人手も限られる組織は導入が進みにくい。こうした状況を踏まえ、既存のゼロトラスト関連製品に焦点を当てると、主に 4 つの課題点が浮かび上がる。

1. ゼロトラストを構成する必須の機能が全て揃った、オールインワンのサービスが少ない。
2. システムの導入に多額の費用が必要となる。
3. ネットワークや複雑なシステムを運用するために高度な知識理解が要求される。
4. 外国製品への依存度が高い。

2. 目的

上記の課題点を解決するため、本プロジェクトにおける主要な目的は次の 3 つとした。第一の目的は、ゼロトラストアーキテクチャの最低要件を満たした、オールインワンのクラウド型プロキシを開発することである。主要機能すべてをクラウド上で利用可能にすることで、ユーザは特段煩雑な導入作業をする必要がなくなり、安全なネットワークを簡単に構築できる環境を提供する。これにより、人的資本の規模に関わらず、多くの組織のセキュリティの向上に繋げることができる。第二の目的は、低コストかつ柔軟性の高いシステムを実現することである。ソフトウェア開発では低レイヤからアプリケーションレイヤまで自前で開発するとともに、サーバやネットワークの構築を自ら行うことで、ブラックボックスとなりうる要素を排し、柔軟性の高いシステムを目指す。同時に、基盤技術から用意することとなるため、外部への依存を低減することが可能である。第三の目的は、必要最低限の知識で利用できる分かりやすいシステムとすることである。具体的には、ルータやファイアウォールといった機器の複雑な操作の習得やポリシー記述言語の習得なしで、誰もが運用に携われ、状態を監視することができる状況を整備する。

3. 開発の内容

本プロジェクトでは令和 4 年にデジタル庁が公表した「ゼロトラストアーキテクチャ適用方針」に適合することを前提とし、次の 4 つのシステムをそれぞれフルスクラッチ開発した。

3.1. ユーザ・デバイス認証基盤システム

ユーザアカウント・デバイス情報を一元的に管理するためのシステムを、ユーザ管理機能、ユーザグループ管理機能、デバイス管理機能を中心に開発した。ユーザ管理機能では、組織内メンバに対してユーザアカウントを発行し管理するとともに、ユーザグループ管理機能として同じ役職等に属するユーザどうしをグループとして管理する。これは主にロール管理の用途で利用することを想定している。

デバイス管理機能では、ユーザが保有する Windows デバイスの情報を取得し、一元的に管理する。デバイス情報を取得するため、Rust 1.82.0 をベースにネイティブアプリケーションを設計・実装した。これを通じて、搭載 OS の詳細情報や接続中のネットワーク情報、Trusted Platform Module の有無、BitLocker によりドライブが暗号化されているかどうか、Windows Hello に対応しているかどうか等のセキュリティ要件情報、位置情報、インストール済みのすべてのアプリケーション情報などを取得する。ここで取得する情報についても、後述するシステムのポリシーの要素として利用することができる。これにより統合エンドポイント管理（Unified Endpoint Management, UEM）の一部機能を代替することが可能である。また位置情報をもとに、ジオフェンシング等に用いることができる。

3.2. アクセス制御ポリシーシステム

ユーザ・デバイス認証基盤システムや外部 IDaaS（Identity as a Service）で管理されているユーザの身元やデバイス情報に基づき、保護リソースに対するアクセスを制御するアクセス制御ポリシーを作成、施行することで安全な接続を実現するシステムを開発した。クラウドサービスへのアクセスを制御する SaaS モード、ファイアウォール等により隔てられ外部からはアクセスすることができない機密性の高い社内システムへのアクセスを制御するトンネリングモードの 2 つを開発し、多様なニーズに応えられる環境を構築した。

SaaS モードでは、SAML 2.0 に基づいて認証を実施し SaaS へのアクセスを制御する。このモードでは、自社のクラウドエンジニアがクラウドサービスにアクセスする際に、クラウドサービスのログイン機能だけでは賄うことができない、高度なアクセス制御を行いたいといったニーズを主に想定している。Google が提供する大規模言語モデル API である Gemini API と連携することで、日本語による自由記述式のアクセス制御ポリシーの作成を実現した。各アプリに対し、必須条件（～を満たさなければアクセスすることができない）と拒否条件（～を満たしたらばアクセスすることができない）を設定することができる。例えば必須条件として「アクセス元の国名が日本であり、所有するデバイスは TPM が有効でドライブが BitLocker により暗号化されていること」という文章を設定することができる。この場合、外国からのアクセスや、セキュリティ要件の低いデバイス所有者からはアクセスさせたくないという場面に対して有効である。認証フローとしては、本システムがいわゆる IdP プロキシとして動作することで保護リソースに対する認証を実行する（図 1）。

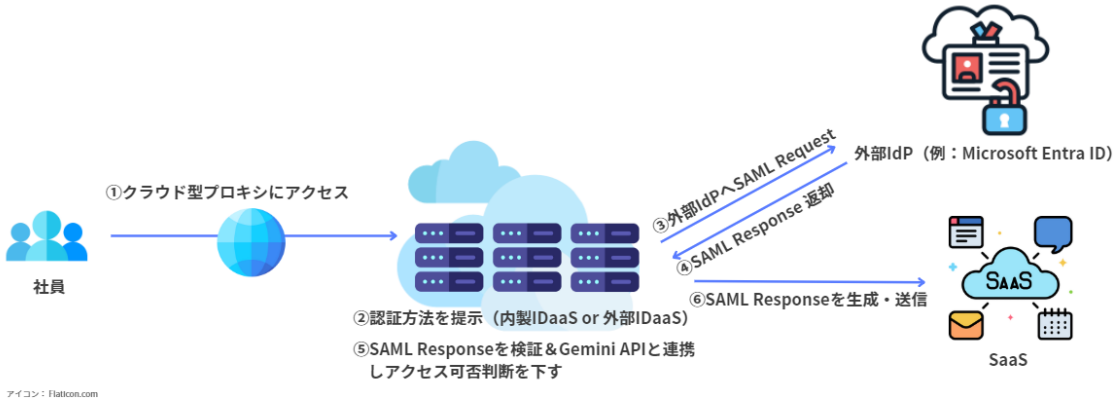

図 1:アクセス制御ポリシーシステム SaaS モードの概要図

トンネリングモードにおいては、社内システムへの安全なアクセスを実現するためのトンネリングサーバとエージェントソフトウェアを Rust 1.82.0 をベースにフルスクラッチで開発した。これらはファイアウォールや NAT 超えを同時に実現することで、導入の際にファイアウォールやルータの設定をする必要がなくなる。またユーザは VPN ソフトウェアをインストールすることなく、ブラウザから簡単にアクセスできる。認証フローは SaaS モードと同じであり、Gemini API と連携した日本語によるアクセス制御が利用できる。エージェントソフトは CLI として実装しており、Rust のクロスコンパイルにより Windows の他 Linux および macOS 等の環境にも対応している。HTTP アプリケーションへのトンネリングを想定しており、独自ドメイン tunneling.me のサブドメインを各トンネルに割り当てることで、ドメインでアクセスできるようにした。加えて、会社等の独自ドメインを用いてアクセスしたいというケースに対応するため、ZeroSSL REST API と連携することで、アプリ内で SSL/TLS 証明書の発行認証から発行・適用までの処理を完結することに成功した。ドメインごとに証明書を発行し割り当てるため、サブドメインのサブドメインなど、ワイルドカードでは保護できない様々な独自ドメイン環境下において HTTPS 化を実現した（図 2）。

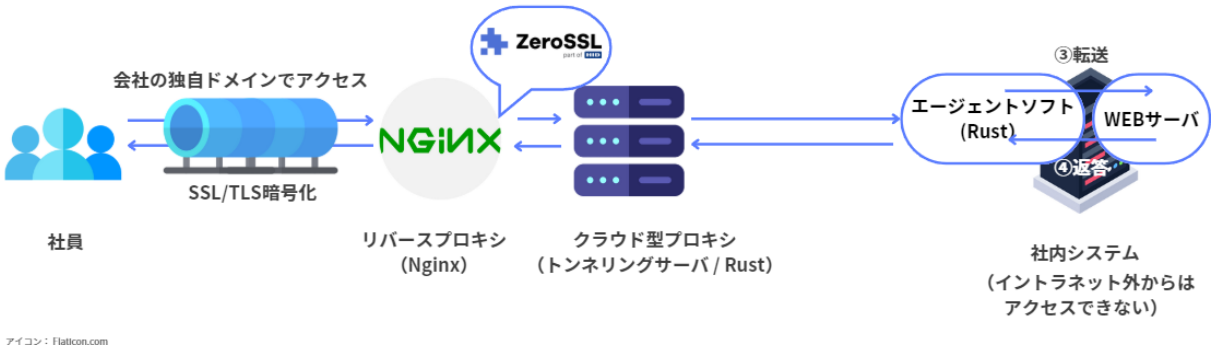

図 2:トンネリングモードにおけるトンネリングシステムの概要図

3.3. インターネット制御監視システム

アクセス制御ポリシーシステムが情報資産の入り口側を保護するアプローチであったのに対し、インターネット制御監視システムはユーザ側からインターネット側へのアクセスをアクセス制御ポリシーにより制御する。Rust 1.82.0 をベースに実装した（図 3）。ポリシーは必須条件と拒否条件を 1 つずつ設定することができ、日本語による自由記述が可能である。

SOCKS5 に対応しているため、デバイスのプロキシ設定に入力することで接続手続きが完了する。「DNS もプロキシを利用する」を選択することで、アクセス先のドメイン名をポリシーの要素として利用することができる。具体的な利用例として、拒否条件に「アクセス先のドメイン名が deepseek.com であること」を設定すれば、いかなるユーザも deepseek.com にアクセスすることはできなくなる。情報漏えいの危険性があるウェブサイトや業務と関係のないウェブサイトにアクセスさせたくないというニーズに応える、ウェブフィルタリングの用途で利用することを想定している。

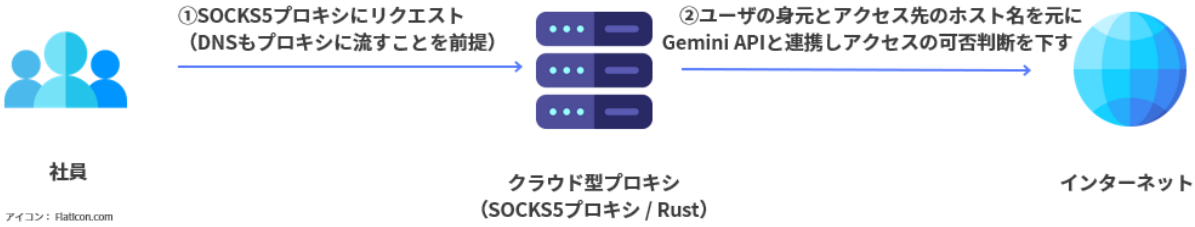

図 3：SOCKS5 プロキシの概要図

3.4. アクセスログ総合管理システム

アクセスログ総合管理システムはアクセス制御ポリシーシステムから得られたアクセスログと、インターネット制御監視システムから得られたゲートウェイログを一元的に管理する。ログとして保管される情報は、アクセス先のアプリ名またはドメイン名（IP アドレス）、アクセス元のユーザ ID、アクセスの許可・不許可情報、その理由、アクセスした際に利用した認証方式情報、アクセス元の国名、発生日時である。許可・不許可となった理由については Gemini API により生成される。またログを保管するだけでなく、収集したログをもとに、動的に新規アクセスが不審かどうかを判断する機能についても実装した。具体的には直近 10 件のアクセスログを随時呼び出し、新規アクセスが疑わしいかどうか Gemini API と連携して判断する。疑わしいと判断された場合は、アクセスが遮断され、当該ユーザに対してメールによる即時通告を行う。この即時通告では、検出したアクティビティの詳細情報のほか、アクティビティが疑わしいと判断された理由を Gemini API が生成し、合わせて通知する。この機能により、先述のシステムではカバーできていない、いわゆる出口対策についてもある程度有効であると考えられる。

4. 従来の技術（または機能）との相違

本プロジェクトの最も特徴的な点のひとつは、ゼロトラストアーキテクチャにおいて重要な役割を果たすアクセス制御ポリシーを日本語で自由に記述できるようにした点である。既存のゼロトラスト製品においては、アクセス制御ポリシーは JSON や YAML、専用のポリシー記述言語など、日常生活で使われていない言語を用いて記述する必要があるため、読み書きできる対象は限られ、人手が足りない組織にとっては大きなハードルであった。当初の目的である、より多くの人がシステムを利用できるようにするという観点から、この機能は有用である。またポリシー記述言語の経験の有無に関わらず、日本語による直感的な操作を可能にしたことで、さらなる利便性の向上が見込まれる。

5.　期待される効果

　ゼロトラストアーキテクチャを構成する必須の機能が揃った、オールインワンのシステムを実現したとともに、ルータやファイルウォール、VPN 等の複雑な操作をはじめとする煩雑な導入作業を低減したことにより、ユーザは各製品の選定や組み合わせの考慮、実際の導入作業等に多くの時間・費用を費やす必要がなくなり、より容易なゼロトラストセキュリティの導入できるが期待される。このようなシステムを通じて、将来的にはイントラネットそのものが必要でなくなる環境を創り上げることができると考える。

6.　普及（または活用）の見通し

　本件はデジタル庁公表のゼロトラストアーキテクチャ適用方針に適合するよう実装に取り組んだため、こうした適用方針に適合する製品の需要は一定存在すると考えている。また、日本国内においては大手の通信事業者であっても自前でゼロトラストセキュリティソリューションを開発する動きはこれまであまり見られず、JANOG54 Meeting に参加した際には、国内からこうした製品を開発することは重要だとお話しされた企業もあり、今後大いに需要があるものと考えている。この先開発や実証実験を進め、通信事業者との提携や事業化についても目指していきたい。

7.　クリエータ名（所属）

小林　麟太郎（The University of Edinburgh, College of Arts, Humanities and Social Sciences）