7X-02

# Webサイトに認証・認可機能を付加するサービスコンテナ

岩原 主† 毛利 公美† 白石 善明‡

岐阜大学工学部電気電子・情報工学科† 神戸大学大学院工学研究科‡

## 1. はじめに

Web サイト運営者は，CMS（Contents Management System）の利用により，容易に Web サイトを構築・運用することができる．すなわち，レンタルサーバによる Web サイトの構築は安価かつ容易であり，サイト運営は技術的な知識が求められなくなっている．会員を限定したコンテンツ配信やそのための会員情報を取得するサイトを運用する際，認証・認可の機能が必要不可欠である．

しかし，Web サイト運営者による認証と認可の機能に対する理解が不十分な場合，それらの機能を Web サイトに実装することは難しいか，セキュリティ上の欠陥を抱えることがあり得る．認証および認可の機能の不備によるサイバー攻撃へ加担する Web サイトを減らすという課題に対して，本研究では認証および認可の機能をレンタル Web サーバの設置のように容易に導入できることを目的としている．本稿では，認証および認可のコンテナを用意し，保護対象の Web サイトにコンテナを接続することで，認証・認可の専門的な知識が無くても，認証・認可の機能を実装できるシステムを提案する．

## 2. 認証・認可機能の既存 Web サイトへの付加

Web サイトに認証・認可の機能を付加するには，Web サーバと連携する認証・認可サーバと認証・認可クライアントが導入される．図 1(a)の Web サイトに単純に導入する場合，モジュールの導入やサーバの設定などの直接的な操作を行うか，レンタルサーバ提供者等により用意された Web 管理画面経由で行うことになる．前者は知識が無い Web サイト管理者には難しく，後者はその実際の操作が隠蔽された代わりに操作誤りや設定不備によるセキュリティの欠陥を抱える可能性がある．図 1(b)のように別のサイト/ホスト/コンテナにある認証・認可サーバを Web サイトのアクセス制御で利用する従来の形態においても，図 1(a)と同様のことになる．

他方で，図 1(c)のように，別のコンテナ，ホスト，サイトに認証・認可クライアントを図 1(b)より疎な形で Web サーバと連携する認証・認可のサーバとクライアントを提供することができれば，既存の Web サイトに対して直接的な操作は不要で認証・認可機能の付加が可能となる．そこで，本稿では，認証・認可サーバ，認証・認可クライアントを Web サーバと分離して設置する方法を提案する．

以上のことに基づいて，既存の Web サイトへの認証・認可機能の付加に求められる要件を以下に示す．

**要件 1**：Web サイト運営者は Web サイト利用者のアクセスを制限可能

認証・認可の機能により既存 Web サイトを保護する．

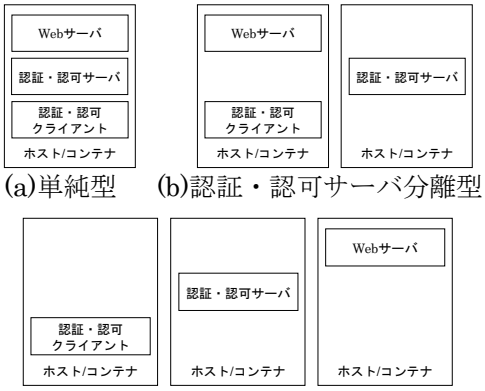


図 1 既存 Web サイトの認証・認可機能の配置から見る本研究のアプローチ

**要件 2**：Web サイト運営者は認証・認可サーバの導入・初期設定が不要

Web サイト運営者に技術的な知識がなくても認証・認可機能を付加できるように，認証・認可サーバの導入・初期設定が不要であることが求められる．

**要件 3**：Web サイト運営者は認証・認可サーバの直接的な操作が不要

Web サイト運営者が Web サイト利用者情報の登録・変更・削除が容易にできることが求められる．

**要件 4**：Web サイト運営者は，Web コンテンツサーバの直接的な操作が不要

Web サイト運営者が認証・認可クライアントの設定を容易にできることが求められる．

## 3. 認証・認可機能の既存 Web サイトへ付加するシステム

提案システムの構成を図 2 に示す．Web クライアントからのアクセスをバックエンドの保護対象 Web サーバに

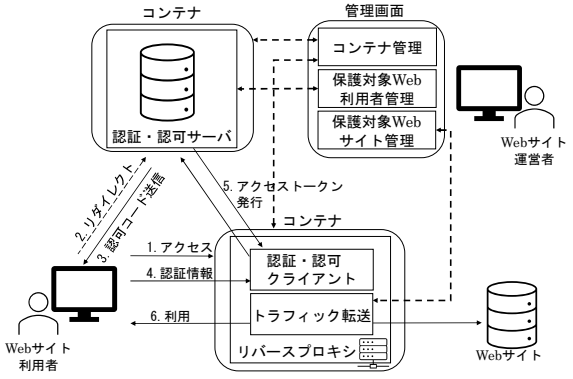


図 2 既存 Web サイトへの認証・認可機能を付加するシステムの構成

---

Service Container Enabling Authentication and Authorization to Websites

† Tsukasa IWAHARA，Masami MOHRI・Gifu University

‡ Yoshiaki SHIRAISHI・Kobe University



Copyright ©2021 Information Processing Society of Japan.
All Rights Reserved.

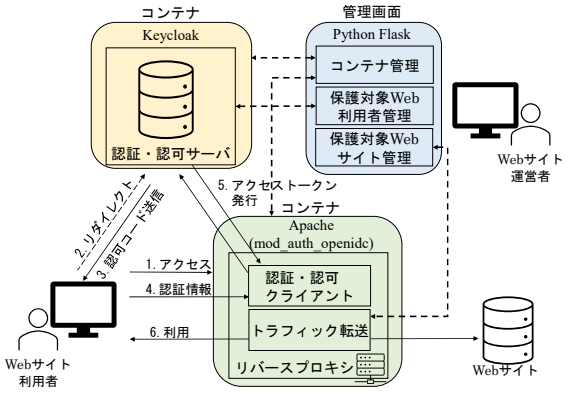


図 3　実装環境

転送できるようなリバースプロキシに Web サーバから分離した認証・認可クライアント機能を実装し，そのリバースプロキシと認証・認可サーバのそれぞれをコンテナとして配置するシステム構成である．

システムの主な機能に，**“コンテナ管理機能”**，**“保護対象 Web サイト管理機能”**，**“保護対象 Web 利用者管理機能”**，**“認証・認可サーバ機能”**，**“認証・認可クライアント機能”**，**“トラフィック転送機能”**があり，システムの動作は次のようになる．

**コンテナ管理**：Web サイト運営者は，認証・認可サーバのコンテナ，リバースプロキシのコンテナの導入，起動，停止，設定変更をブラウザの画面から行う．【要件 2 に対応】

**保護対象 Web サイト管理**：Web サイト運営者は，保護対象 Web サイト URL の登録，変更，削除をブラウザの画面から行う．保護対象 Web サイトの URL の登録を行うと，リバースプロキシのバックエンドとして接続する Web サイトの設定が完了する．【要件 4 に対応】

**保護対象 Web 利用者管理**：Web サイト運営者は，認証・認可サーバのデータベースに Web サイト利用者の名前，パスワード，メールアドレスの登録，変更，削除をブラウザの画面から行う．【要件 3 に対応】

**認証・認可サーバおよびクライアント**：認証・認可サーバは，保護対象 Web サイトにアクセスしたい Web サイト利用者の認証を行い，バックエンドの Web サイトへのアクセスを許可する．【要件 1 に対応】

## 4.　実装

図 3 に実装環境を示す．認証・認可サーバは Keycloak[1]を使用する．リバースプロキシは，Apache を使用し，mod_auth_openidc モジュール[2]を認証・認可クライアントとしてインストールする．Web サイト運営者の管理画面は，Python の Flask フレームワーク[3]により提供される．図 4 は，コンテナの管理画面である．Docker[4]によりコンテナ環境を構築している．図 5 の画面は，Web サイト利用者として登録されている利用者情報が表示され，この画面から登録，変更，削除の操作を行うことができ，いずれかの操作を行うと Keycloak に Web API を用いて反映される．図 6 は，保護対象 Web サイトの URL 管理画面であり，この画面からリバースプロキシのバックエンドにつなぐ Web サイトを設定する．


図 4　コンテナ管理画面


図 5　保護対象 Web サイトの管理画面


図 6　保護対象 Web 利用者の管理画面

表 1　Keycloak による図 1(b)の形態の実装と提案システムの比較

| | Keycloak | 本提案 |
|---|---|---|
| 要件 1 | ○ | ○ |
| 要件 2 | − | ○ |
| 要件 3 | − | ○ |
| 要件 4 | − | ○ |

## 5.　まとめ

本稿では，既存の Web サイトに認証・認可の機能を容易に付加できるシステムを提案した．

Keycloak を用いて図 1(b)の形態を実装した場合と本提案システムで認証・認可の機能を付加する方法の違いを表 1 にまとめる．提案システムでは，コンテナの設置により，Web サイト運営者が導入・初期設定を行う必要がなく（要件 2），認証・認可サーバへ Web サイト利用者の登録をブラウザの画面から容易に行うことができる（要件 3）．また，本提案はリバースプロキシ型の構成により，サイト運営者が Web サーバに直接操作を行わずに認証・認可の機能を付加することを可能にした（要件 4）．

**謝辞** 本研究は JSPS 科研費 JP18K04133，JP19K11963 により行われた．

## 参考文献

[1] Keycloak，https://github/keycloak/keyclaok
[2] mod_auth_openidc，https://github.com/zmartzone/mod_auth_openidc
[3] Flask, https://flask.palletsprojects.com/en/1.1.x/
[4] Docker, https://www.docker.com/



Copyright ©2021 Information Processing Society of Japan.
All Rights Reserved.