記事
Toshihiko Minamoto · 2022年3月24日 4m read

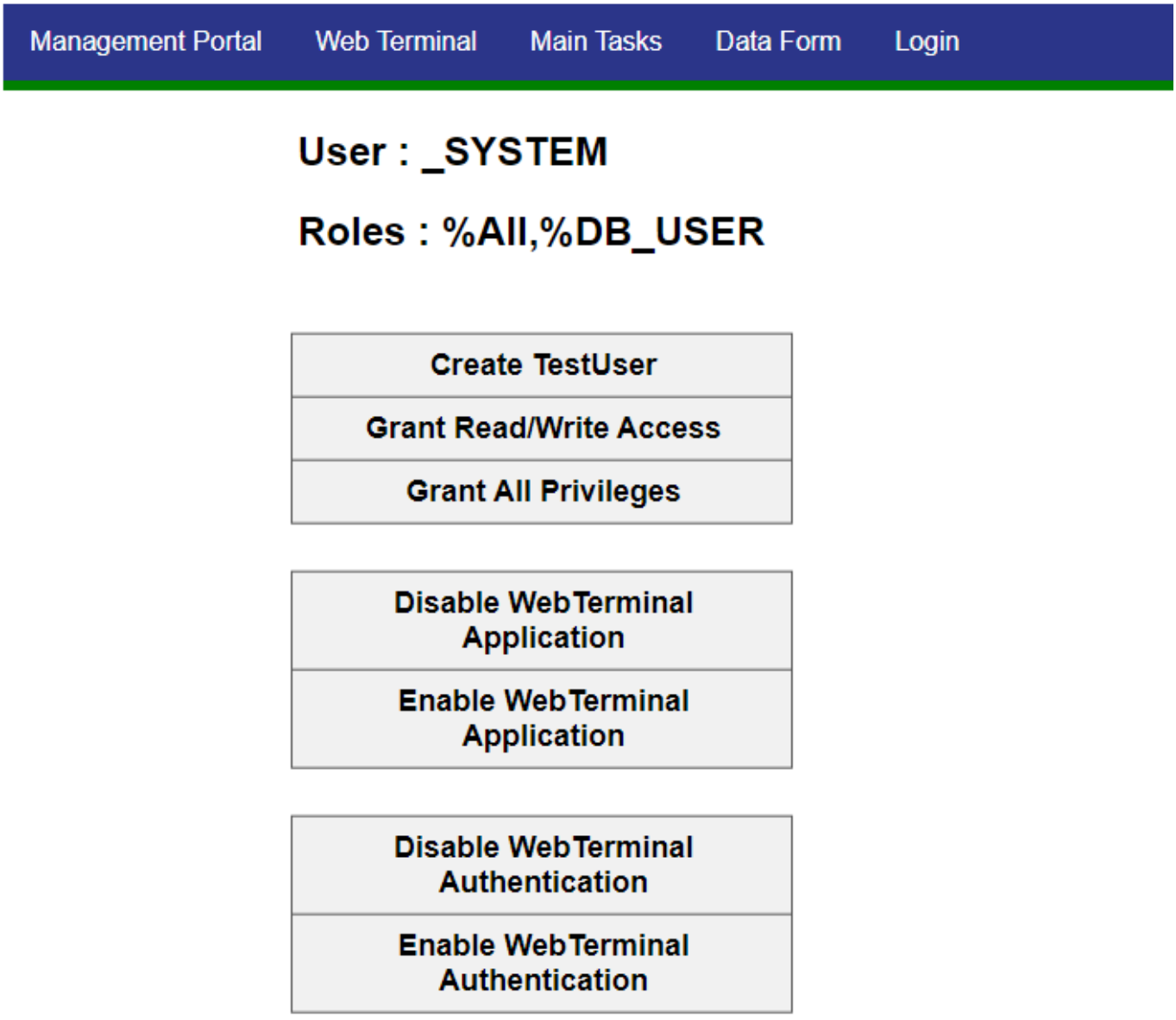
Open Exchange

Web アプリケーションからのコードによる OAuth2 と基本認証、承認、および監査

この記事では、CSP Web アプリケーションを使用して、コードで認証、承認、および監査を行う方法と、Web アプリケーションを有効化/無効化および認証/認証解除する方法について説明します。

- オンラインデモは、クラウド（https://dappsecurity.demo.community.intersystems.com/csp/user/index.csp ）にあります（SuperUser ¦ SYS）。
- 先に進む前に、動画をご覧になることをお勧めします（https://www.youtube.com/watch?v=qFRa3njqDcA）。

アプリケーションのレイアウト



認証から始めましょう

認証は、InterSystems IRIS® に接続しようとするユーザーや他のエンティティの ID を検証します。
よく言われるように、認証は、ユーザーがユーザーが言う通りの本人であることを証明する方法です。

ユーザーを認証する方法にはいくつかあり、それぞれが認証方法として知られています。 InterSystems IRIS では、以下のような多数の認証方法をサポートしています。

- Kerberos — Kerberos プロトコルは、セキュリティで保護されていないネットワークを使用するサービスに安全な認証を提供するために設計されました。 Kerberos はユーザーの認証にチケットを使用し、ネットワークでのパスワードの交換を回避しています。
- オペレーティングシステムベース — OS ベースの認証は、各ユーザーのオペレーティングシステムの ID を使用して、InterSystems IRIS に対してユーザーを識別します。
- インスタンス認証 — インスタンス認証では、InterSystems IRIS はユーザーにパスワードを求め、提供されたパスワードのハッシュと InterSystems IRIS が保存している値を比較します。
- LDAP（Lightweight Directory Access Protocol） — LDAP では、InterSystems IRIS は LDAP サーバーとして知られる中央レポジトリにある情報に基づいて、ユーザーを認証します。
- 委任認証 — 委任認証は、カスタマイズされた認証方法を作成する方法を提供します。アプリケーション開発者が、委任された認証コードの内容を完全に制御します。

私は、インスタンス認証を使用しています。ユーザー作成には、以下の ObjectScript コマンドを使用できます。

```
  &sql(CREATE USER TestUser IDENTIFY BY demo)
```

demo パスワードを使用する TestUser を作成しました。

監査

ユーザーを作成すると、そのレコードは、以下の ObjectScript コマンドを使用して監査データベースにも追加されます。

```
Do $SYSTEM.Security.Audit("%System","%Security","UserChange","User:TestUser | Passwor
d:demo","Audit Log inserted from Data_APP_Security")
```

次の関連ドキュメント（監査ガイド）をお読みください:
https://docs.intersystems.com/irislatestj/csp/docbook/DocBook.UI.Page.cls?KEY=AAUDIT

承認

認証が完了したら、ロールを作成してそのロールに権限を付与し、ロールとユーザーをリンクする必要があります（承認）。 これは、3 段階で行います。

ステップ 1: 以下の ObjectScript コマンドを使用して、ロールを作成します。作成するロールは ReadWrite ロールです。

```
&sql(CREATE ROLE ReadWrite)
```

ステップ 2: テーブルに対する SELECT、UPDATE、INSERT 権限をロールに付与します。scw.Patient テーブル権限を ReadWrite ロールに割り当てます。

```
&sql(GRANT SELECT,UPDATE,INSERT ON scw.Patient TO ReadWrite)
```

ステップ 3: ロールをユーザーに付与します。ReadWrite ロールを TestUser ユーザーに割り当てます。

```
&sql(GRANT ReadWrite To TestUser)
```

Web アプリケーションの有効化/無効化

以下の ObjectScript コードを使用して、Web アプリケーションを有効化または無効化することができます。

```
New $Namespace
Set $Namespace = "%SYS"
Set App = ##class(Security.Applications).%OpenId("/terminal")
Set App.Enabled=0
Do App.%Save()
```

ここで「/terminal」はアプリケーションの名前です。 アプリケーションは、「App.Enabled」を 0 に設定することで無効化、それを 1 に設定することで有効化することができます。

Web アプリケーションの認証/認証解除

以下の ObjectScript コードを使用して、認証を設定できます。

```
New $Namespace
Set $Namespace = "%SYS"
Set App = ##class(Security.Applications).%OpenId("/terminal")
Set App.AutheEnabled=0
Do App.%Save()
```

ここで「/terminal」はアプリケーションの名前です。
認証は、「App.AutheEnabled"」プロパティを使用して設定できます。 以下の数値を設定可能です。

```
property AutheEnabled as Security.Datatype.Authentication [ InitialExpression = 64 ];

Authentication and Session mechanisms enabled (CSP Only).
Bit 2 = AutheK5API
Bit 5 - AutheCache
Bit 6 = AutheUnauthenticated
Bit 11 = AutheLDAP
Bit 13 = AutheDelegated
Bit 14 = LoginToken
Bit 20 = TwoFactorSMS
Bit 21 = TwoFactorPW
```

以上です！

#InterSystems IRIS
InterSystems Open Exchangeで関連アプリケーションを確認してください

ソースURL:
https://jp.community.intersystems.com/post/web-%E3%82%A2%E3%83%97%E3%83%AA%E3%82%B1%E3%83%BC%E3%82%B7%E3%83%A7%E3%83%B3%E3%81%8B%E3%82%89%E3%81%AE%E3%82%B3%E3%83%BC%E3%83%89%E3%81%AB%E3%82%88%E3%82%8B-oauth2-%E3%81%A8%E5%9F%BA%E6%9C%AC%E8%AA%8D%E8%A8%BC%E3%80%81%E6%89%BF%E8%AA%8D%E3%80%81%E3%81%8A%E3%82%88%E3%81%B3%E7%9B%A3%E6%9F%BB