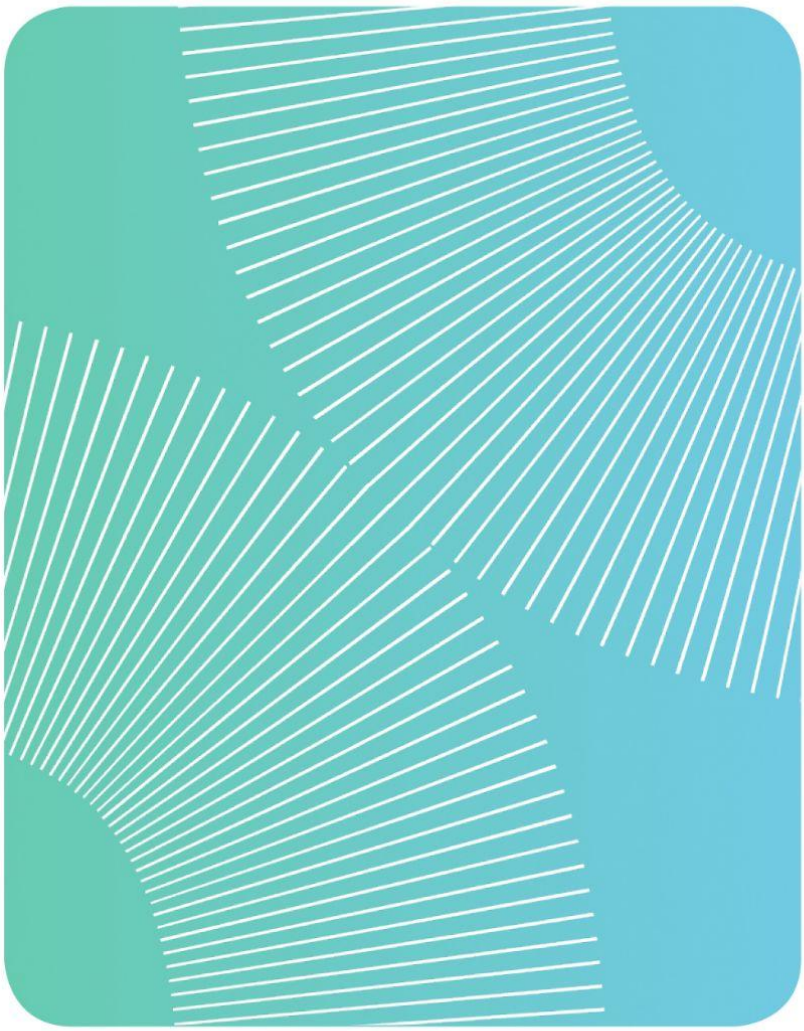
auth0

Best Practices Guide for SMART on FHIR



GUIDE

auth0.com

Introduction

The SMART on FHIR Framework unites healthcare workers and their patients by harnessing the latest technology standards in order to connect clinical documentation, patient-clinician communication, billing, and the Electronic Health Record (EHR).

The Fast Healthcare Interoperability Resources (FHIR) Standards Framework expands the interoperability of healthcare technology. Information sharing between third-party applications and the EHR is now possible even between different platforms through RESTful practices.

The SMART Application Launch Framework is a reliable authorization protocol that supports any FHIR system. The FHIR scope covers many architectural scenarios to support integrated healthcare, from clinical trials to patient billing. Best of all, it's easy to implement, built on the latest standards, and leverages Clinical Documentation Architecture (CDA).

FHIR is essentially a collection of healthcare instances called “resources” that can be accessed based on a given institution’s policies. FHIR is a comprehensive foundation for different healthcare contexts, from clinical decision making to financial reimbursement for services. The FHIR API is the definitive patient access API and provides developers readability from data to endpoints. Particularly, the Da Vinci Payer Data Exchange (PDex) showcases powerful interoperability of FHIR resources and payer data alongside financial and clinical requirements.

However, although the data modelling is impressive, complex use case implementation can quickly become an infrastructural nightmare, and security protocols are left to the developer to manage.

Auth0 provides a secure authorization and authentication platform that is compatible with SMART applications, and enables customers to make access requests to FHIR resources in the form of OAuth 2.0-compliant authorization through Auth0 Tenant Servers and authentication through the OpenID Connect standard. Auth0 offers HIPAA Business Associate Agreements to customers handling PHI data. Using Auth0 as an identity and authentication service helps companies to be HIPAA-compliant.

SMART vs OAuth2

FHIR is a standards framework for exchanging healthcare data, and the SMART Launch Framework secures the FHIR API so that only authorized users/services can access that data. SMART may share some similarities with OAuth2, but there are a few key differences:

1. The SMART authorization service is more flexible when it comes to determining an active patient record or providing user consent for viewing multiple patients.
2. SMART provides session context next to a given Access Token response. This does not happen with OAuth2; instead, OAuth2 relies on the OIDC protocol to communicate application session-level information.
3. SMART has additional authorization server metadata endpoints that are not part of OAuth2.
4. SMART supports public application authentication via a single shared key for all public SMART clients.

This best practices guide covers how to address each of these differences with a PDex scenario of a clinical portal, Clinic0, connecting a patient, Joe Smith, to a third-party Insurance website, Med0, to share Joe's visit history and his carrier coverage options.

Stateless

SMART apps should be stateless—that is, request transactions are independent of session context. Session-based and browser-based data should be stored in separate cookies. This cuts down on memory usage and eliminates sticky server-side sessions, ultimately supporting scalability.

And—if this isn't already obvious—in stateless applications, never include bearer tokens in cookies 🤗.

Clinic0 doesn't currently have a way to remember patient selection, but its developers are working on incorporating DynamoDB and TTL in order to maintain previously selected data, with one Access Token associated per patient.

Authorization & Authentication Parameters

Although it is not required in V1 of SMART on FHIR standards, PKCE authorization with Auth0 supports secure interaction with the FHIR API. Otherwise, the application will need an additional endpoint to authorize with Auth0.

For SMART apps, state (`state`) and audience (`aud`) parameters must be included with the authorization request. State is a unique and opaque parameter to prevent phishing and CSRF attacks. The audience parameter should be the URL of the FHIR resource server from which the app wishes to retrieve FHIR data in order to prevent leaking a genuine bearer token to a counterfeit resource server.

Again, though not strictly required by the SMART Application Framework, OIDC specifications indicate that a more-secure method to authenticate a requesting party is to treat the user claim/represent the user as the URL of an FHIR resource, in addition to a pair of OIDC scopes.

SMART apps should always use a restrictive scope when requesting access to a given FHIR resource.

```
{
  "access_token":"XXXX.",
  "aud": https://yourfhirresource.io/r/12345,
  "scope": "openid fhirUser launch patient:read”,
  "state": "xyzABC123”,
  "fhirUser": "https://yourfhirresource.io/r/12345"
}
```

In our B2C use case, Clinic0, built on the SMART Patient Standalone Launch SDK, gives Joe the ability to link Med0 as his insurance carrier.

Consent

Consent is an integral element of SMART applications. There are different types of consent authorizations which vary based on context, from consent to share personal information to permitting authorized representatives to access patient records. After a given user is navigated to your login page by your Auth0 Authorization Server to authenticate, SMART applications must request consent in order to access FHIR resources on the user's behalf.

Clients created in the Auth0 dashboard are assumed to be first-party applications. Since consent legislation varies per region, first-party application consent is optional; however, we recommend setting `Allow Skipping User Consent` to disabled. In the case of third-party applications, remember to set `is_first_party: false` in the management API. It is imperative that SMART apps include a consent flow to accommodate IIHI scenarios according to HIPAA regulations.

Clinic0, the FHIR resource owner, uses an Auth0 Redirect Rule to connect Joe to Med0, a third-party application. The Rule redirects Joe to a custom consent screen which prepends Joe's patient identifier and the appropriate scopes. If Joe does not accept the consent screen, Clinic0 does not connect Joe's insurance payer (Med0) information.

Claims

SMART launch specs require session information or launch context parameters such as patient ID (`patient_id`) to be returned next to the Access Token. Since this is not out of the box with OAuth2, one option available with Auth0 is to create a Rule to modify the Access Token and add custom claims like `patient_id` so that these claims are included alongside the Access Token in the response.

Your SMART app can then use this Access Token to interact with the FHIR API. Rules run during every authentication, which means the Access Token may include any type of claim, not only a patient parameter but for example, encounters or observations. Remember to use a unique name, and take advantage of URL namespacing.

SMART apps should use a minimal number of claims for authorized users. Do not include sensitive patient information in the payload, and make sure to stick to SMART requirements and keep Access Token scope manipulation to a minimum.

Clinic0 sends custom claims which are limited to Joe's patient ID, his permissions via scope, and Med0 as the audience.

Proxies

Sometimes, use case requirements go beyond custom claims in the Access Token. Since Auth0 doesn't support custom claims in an HTTP response out of the box, Auth0 customers with complex application requirements can take advantage of services such as AWS CloudFront and Auth0 Private Cloud on AWS. Lambda functions can act as proxies and send requests to Auth0's `/authorize` and `oauth/token` endpoints so that your custom namespaced claims are in the issued response.

Clinic0 uses a proxy to connect Joe's Clinic0 visit history with his Med0 payer information. Clinic0's developers followed a security-forward design pattern in order to avoid a single point of failure.

Authorization Policies

Access for a given patient, study, or clinical instance has many granular security implications. To some extent, role-based authorization of SMART clients may be achieved through Auth0's Core Authorization, which includes policy evaluation based on roles and permissions assigned to users.

Clinic0 uses role-based access control (RBAC) for basic application navigation, and access control lists for more fine-grained authorization administration and management.

Once you've enabled RBAC, you can further customize your authorization policy by using Rules.

Clinic0's patient portal not only connects their patients' third-party insurer information, but also grants their patients access to their pharmacy prescriptions, upcoming appointments, and lab results. Clinic0 also has a variety of clinician-dedicated SMART

applications for risk assessment tracking, such as blood pressure and cardiac health. All of these apps, including the patient portal, leverage data from one another, from demographics to lab results. This illustrates the potential of interoperability with SMART on FHIR, and with great interoperability comes great responsibility. Because the FHIR API ultimately handles population health, it is imperative that individual SMART apps have their own authorization policy, as well as an authorization policy for your organization.

Public Registration

Having already discussed PKCE authorization, Auth0 also has the added security benefit of automatically exposing a JSON Web Key (JWK) endpoint for each Tenant, which makes possible a single shared key for all public SMART clients, as per the SMART backend service registration specification.

Clinic0's JWK is used to sign all incoming JWTs.

Confidential Registration

Auth0 supports confidential client registration using client secrets, securing both web and mobile applications.

Refresh Tokens

Refresh token rotation for Single Page Applications (SPAs) is a powerful Auth0 differentiator.

Developers at Clinic0 are working on taking advantage of refresh tokens for their SMART Application, testing out their new feature set with FHIR Public Test Servers.

Unsupported Use Cases

While the future looks bright, session timeouts, updates, and SPA refresh are not currently covered in the SMART application profile context.

Conclusion

Despite the fact that Clinic0 is a fictional developer scenario, BioReference Laboratories handled a 25x volume spike during onset of the COVID-19 pandemic with Auth0 at the helm, supporting their patient portal experience which integrates FHIR, enabling patients to link their data between platforms.

“I don't ever have to worry about security or user access. I have Auth0,” said Vice President of Consumer Technology and Digital Solutions Vinny Pacione.

If you are working on bridging your application to other SMART on FHIR applications and want the best in authorization and authentication, please reach out to our Auth0 experts to get started today.

Auth0’s modern approach to identity enables organizations to provide secure access to any application, for any user. The Auth0 Identity Platform is a highly customizable identity operating system that is as simple as development teams want, and as flexible as they need. Safeguarding billions of login transactions each month, Auth0 delivers convenience, privacy, and security so customers can focus on innovation.

For more information, visit **https://auth0.com**