Maximise Security by Adding SMART(s) to your FHIR APIs

Orion Health White Paper
Dr. David Hay
Product Strategist
082016

Supporting safe access to data, enabling the open healthcare ecosystem

FHIR®, or Fast Healthcare Interoperability Resources, is one of the next generation HL7® standards in healthcare data integration. It focuses on decreasing interoperability costs, and unlocking technical innovation in healthcare by supporting an open ecosystem of information providers and consumers via open APIs. But with any API and particularly one that exposes Personal Health Information (PHI) there will be security issues to consider.

So now there is a new acronym SMART (standing for Substitutable Medical Applications and Reusable Technologies) that is also generating excitement within the community. SMART adds a layer of security in front of FHIR interfaces to support safe access to data held within an EHR – or any other repository.

Focused on implementers, FHIR reuses many of the concepts already familiar to developers from other domains. These include Resources to represent common healthcare concepts such as Allergies, Medications and Problems. This enables customisation of these resources for specific uses (Profiling) and a simple REST based API made popular by some of the major internet players such as Google, Twitter and Facebook, which support both XML and JSON. FHIR has the support of many of the large healthcare organisations and vendors, as well as national bodies such as ONC in the United States, the NHI in the United Kingdom and NEHTA in Australia.

SMART is not yet as well-known as FHIR, but healthcare organisations and national bodies are taking an active interest in its development, through projects such as Argonaut. SMART leverages the existing standards OAuth2 for Authentication and Authorisation, OpenID Connect for user Identity and standardises the process of negotiating access to information and operations between app and server. It also describes a process by which an EHR application can launch an external app preserving context (patient and user), and providing safe access to the data within the EHR or, indeed, any other repository of healthcare data.

This paper reviews SMART, and considers how an organisation can support and benefit from this new healthcare standard.

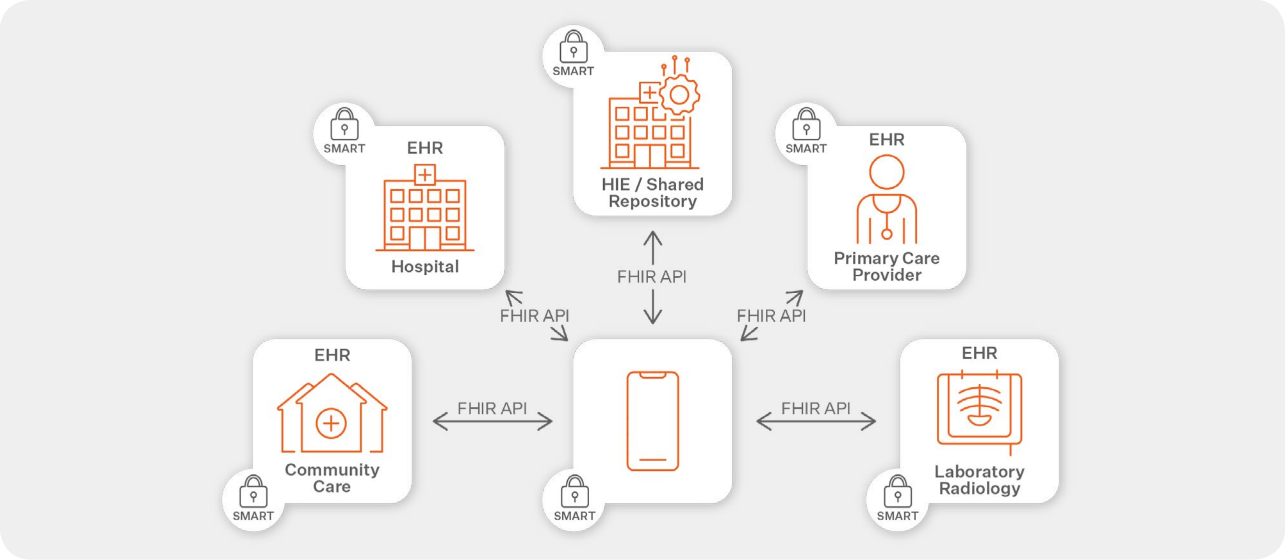

FIGURE 1: FHIR APIs Protected by SMART

Details on SMART

SMART was originally proposed to enable specialist external applications (applets or apps) to interact securely with the data in an EMR system. The vision was by standardising the communication between the two, both in terms of API and content, it would be possible to create an 'ecosystem' of apps and data sources. A single app could interact with multiple EHRs in the same way as mobile applications can interact with on-device resources, such as contacts lists, phone and camera.

When first described SMART used its own definitions for content and API, but once FHIR gained traction, it switched to use FHIR for both of these.

The problem SMART is trying to solve is that there are large amounts of clinical information collected within EMR systems, that is only available to the EMR vendor. Any application development that uses this data could only be done by the application vendor. This reduces the ability to develop specialised and innovative applications. By splitting the roles of data consumer and data supplier, it becomes possible for vendors to develop specialist applications that run against different data sources – something called 'sidecar' applications by Gartner analyst Wes Richell. See blog post: https://fhirblog.com/2015/07/10/fhir-smart-and-sidecar-applications/

However, since its original description, SMART has become a more general standard for securing access to systems exposing FHIR resources:

SMART focuses on three key areas – leveraging other standards to do so:

- Providing a way for a client application and user of that application to identify themselves and to negotiate access to the data in the EHR. This leverages the OAuth2 standard.
- Representing the information that is being exchanged, and the manner in which it can be accessed. This uses FHIR Resources (specifically utilising FHIR profiles for the details) for the content, and the FHIR REST API as the query mechanism.
- Providing a mechanism by which an external app could be 'launched' from an EHR – preserving the context (current patient and user), or how it can access the EHR information safely from an external call.

Identifying the client and negotiating access

There are a number of 'roles' that are important in understanding SMART (these come from the underlying OAuth2):

- The Resource Provider is the supplier of the data – the system that wishes to make the data available via FHIR interfaces to a client.
- The Authorisation Server is the component that identifies and authorises (determines what they can access) the client app and the user wishing to access the data. It could be part of the same application as the Resource Provider, or a separate one.
- The App is the actual application that is requesting the information (e.g. a smart phone app). It is also called a client.
- The User is the person who is using the app. (Note that there is a difference between the User, just the entity that supplied the login details and the details about that user: name, date of birth, address etc.). The latter refers to the user 'Identity', and is itself an optional scope as described below).

While the implementation details are complex (and can vary from site to site, even within the SMART standard) from a high level it is straightforward. A pre-requisite is registering the app with the Authorisation Server, which supplies it with an identification key.

When the user uses the app to request some data:

1. The Authorisation Server checks that it recognises the app (the app has to supply the key that identifies it). This allows the Authorisation Server to customise its behavior based on the capabilities of the app.
2. The Authorisation Server identifies the user of the app by requesting a Username/Password, or some other mechanism that the Authorisation Server trusts.
3. Next the Authorisation Server determines what data, and functions on that data (read, insert, update, delete) the user can perform. The actual details of how this occurs varies according to the implementation. This is expressed as the 'scope' of the access. The app can request a particular set of scopes, but the Authorisation Server is not bound to permit all (or any) of these. Examples of scope could be: read access to all patient data, the ability to select a new patient, or disclosing the identity (name etc.) of the user of the app.
4. Having identified the user (simply that they are a valid user – not necessarily the 'Identity' details) plus determined which data they can access (the scope), the Authorisation Server then issues the app with a token (called the Access Token) that the app can then use when requesting the data.
5. Finally, the app makes one or more FHIR requests against the Resource Server, including the Access Token in the request. The Resource Server checks that the token is valid and that the user can make

Details of the process can vary according to the specific implementation. For example, the Authorisation Server could place the agreed scope in the Access Token, and encrypt then sign it, or the Access Token could simply be a random key indexing an entry in the Authorisation Server, and the Resource Server uses it to check each request with the Authorisation Server as the diagram above shows. SMART will work regardless of the server's implementation details, which is the whole purpose of the standard.

The API and the Data content

SMART provides another advantage, related to the data content – the FHIR resources. While FHIR provides descriptions of each resource (for example, what information is inside an Allergy), it does so in a general way. Specific implementations then 'adapt' that resource to meet their specific requirements, a process called 'profiling' FHIR. Although not necessary for interoperability, it is helpful that both client and server recognise the same profile therefore SMART defines a standard profile.

In fact, this aspect of SMART is undergoing development at the moment, currently there is a 'SMART profile', but in the US this is moving towards using the Data Access Framework (DAF) profile. There is also work on adapting DAF for other countries, for example Canada, so it is likely that SMART will allow a client to include the profile/s it needs as part of the negotiation process.

Launching an app

SMART describes a couple of ways of launching an app. From within the EHR (the EHR launch) and by an externally launched app (Standalone launch). The standalone launch is simply an external app requesting access via the API as described above.

The 'EHR launch' describes how the EHR can start a SMART aware app, preserving the current EHR context i.e. the current user and patient (if selected).

The workflow is:

1. The user invokes a function to launch a previously registered app (e.g. clicks on a link in the User Interface).
2. The EHR stores the context information, creating a token (the launch token) that refers to it. Alternatively the EHR could include the context within the token encrypting and signing it.
3. The app then authenticates itself and the user in the same way as described above.
4. After authentication, the app includes the launch token with each call to the Resource Server. The Resource Server uses the token to retrieve the context (patient and user) of the call, either by using the token as a key to some internal store or by decrypting it – depending on how it was initially created in step 2.

Note that like the other aspects of SMART, most of the server-side implementation details can vary between implementations. If the interface between the client and server is preserved, it will have no effect to the client.

Healthcare Organisations and SMART

Where are the areas that SMART could be considered and benefit a healthcare organisation?

Protecting FHIR based interfaces

APIs in general are a way of 'future-proofing' an organisations investment in data and access to that data, and in the healthcare world FHIR is really the only option at this time. SMART uses FHIR, so therefore it is necessary that the APIs being protected are conformant to the FHIR standard. SMART is a good choice for providing this security, especially as it is based on OAuth2, a very common standard used in health and other domains and is familiar to developers.

The specific steps organisations would need to do to support SMART are:

- To recognise and support the OAuth2 scopes that define the data access allowed. Note that this does not replace the current privacy mechanisms as these will always be applied internally regardless of the access described in the scope (and permitted) by the Authorisation Server.
- To recognise and support the specific FHIR profiles, such as DAF.

Supporting SMART aware clients

The EHR Launch mechanism could be utilised in several ways:

- It describes a useful abstraction for development. If the healthcare organisations EHR were to support this capability (in conjunction with access to data via the FHIR API) then it would decouple the EHR development from app development. Indeed, using this framework could provide a means of rapidly developing 'EHR-like' capability that could be customised for a particular implementation.
- As the use of SMART continues to grow, it is likely that there will be an increasing number of SMART compliant applications available - this is occurring currently in the genomics space. Supporting these applications will allow healthcare organisations to develop population health management and precision medicine capabilities, by utilising existing applications rather than having to develop their own.

Future growth of SMART

SMART (like FHIR) is an evolving standard, and so is likely to change and extend as implementations occur. The areas where we are likely to see evolution are:

- Becoming the 'de facto' security standard for FHIR interfaces. FHIR itself doesn't prescribe a security mechanism, leaving that to the implementer. Given SMARTs use of widely accepted standards, it is likely that it will become the preferred security mechanism.
- Supporting different profiles. As described in this paper, SMART is already moving to allowing the client to describe the profiles it supports.
- CDS Hooks (Clinical Decision Support), is an exciting new development that seeks to standardise how an EHR can invoke Decision Support capability in the course of its usual operation. This standard, if it becomes widely adopted, could make it easier for a provider of Decision Support services to be utilised by different EMR/EHR systems – another important aspect of exposing and consuming advice generated as part of the precision medicine initiative.

Conclusion

This paper describes how SMART provides an access and security layer on top of the next generation FHIR integration standard by utilising the existing and widely implemented OAuth2 security standard.

FHIR and SMART working together provides secure and safe access to data held within an EHR, or any data repository using a well-known API managed by the custodian of the clinical data. With the growing support for SMART by large healthcare organisations, vendors, providers and national bodies, this will promote free flowing healthcare information that in turn can lead to different 'specialist' applications. These applications, each focused on some aspect of health care delivery can access data from different data sources, creating numerous 'sidecar' applications, truly enabling the open healthcare ecosystem.

Dr. David Hay - Biography

David is a Product Strategist for Orion Health. He is also active in the international standards community as the chair of HL7® New Zealand and is a co-chair of the FHIR® Management Group, charged with guidance and development of the latest HL7 standard. David graduated from medical school, then moved into the Health IT sector. He started a company developing Practice Management Software to GP's (GPDat). This was the first such program in New Zealand to receive electronic laboratory results.

Leaving the vendor space, David worked at EDS as a Solutions Architect, before returning to health IT working as an Architect (Solutions and Enterprise) for the Auckland based services management organisation 'healthAlliance' producing a number of innovative solutions: an internal eReferrals application, a community based Case Management solution, plus an application to track and report on clinical tasks within the hospital. All of these applications remain in current use today. While there, he also participated in a number of national programs, including the medical records transfer project GP2GP, ePrescribing, and was one of the authors of the Interoperability Reference Architecture that governs information flow within the health sector.

He currently serves on the Health Information Standards Organisation (HISO) committee, which provides technical advice on standards to the New Zealand Digital Advisory Board. He has provided outstanding leadership towards helping to create and evangelise the innovative FHIR standard, and frequently writes about FHIR on his blog fhirblog.com and has developed a range of open source tools widely used to educate and assist the developers of FHIR clinfhir.com.

David is a true champion of the New Zealand IT healthcare sector, he has spent many years both professionally and personally designing, educating, and advising on health informatics. In recognition of this David has been awarded the Excellence in Health Informatics award 2016 by ITx New Zealand. His ongoing commitment to FHIR has helped to provide the momentum needed for international adoption. FHIR represents a major standards upgrade that will boost access to health information, which will improve the access to a patients health information globally.

ORION HEALTH