The Office of the National Coordinator for Health Information Technology

KEY PRIVACY AND SECURITY CONSIDERATIONS FOR HEALTHCARE APPLICATION PROGRAMMING INTERFACES (APIS)

Prepared on behalf of the U.S. Department of Health and Human Services (HHS), Office of the National Coordinator for Health Information Technology (ONC) under Contract: HHSP233201600224A, ESAC, Inc. and SRS, Inc.

December 2017

Table of Contents

Figures

Tables

EXECUTIVE SUMMARY

This document describes key considerations for implementing and managing application programming interfaces (APIs) in healthcare with respect to the privacy and security of health information (e.g., electronic protected health information (ePHI)).[1] These considerations were developed as a result of testing and assessing a volunteer subset of the implementations of the Sync for Science (S4S) API in accordance with applicable Precision Medicine Initiative (PMI) Privacy and Trust Principles (PMI Privacy Principles)[2] and the PMI Data Security Policy Principles and Framework (PMI Security Principles).[3] Special publications from the National Institute of Standards and Technology (NIST) also served as a basis for assessment criteria of the participating S4S pilot organizations.[4] Entities covered by the Health Insurance Portability and Accountability Act (HIPAA) must implement appropriate privacy protections and data security safeguards in their environments, and in particular, comply with the HIPAA Privacy and Security Rules.[5] The PMI Privacy and Security Principles are consistent with HIPAA and can help bolster an entity's privacy and security posture.

The use of APIs in healthcare, which can enable individuals (e.g., patients or their personal representative) to request that a healthcare provider's electronic health record (EHR) send health information about them to a specified third-party, such as a research application (app), can leverage the below considerations to help ensure privacy and security of health information with the appropriate safeguards in mind.[6]

KEY PRIVACY CONSIDERATIONS

The following are key areas for privacy consideration when implementing APIs in healthcare:

1. Ensure that any electronic access request interface (e.g., an electronic form within a patient portal) provides individuals with an opportunity to approve the electronic transmission of health information in accordance with applicable legal requirements that enable such a request (e.g., HIPAA right of access).

---

[1] https://www.hhs.gov/sites/default/files/ocr/privacy/hipaa/administrative/combined/hipaa-simplification-201303.pdf

[2] The *PMI Privacy Principles* document is a set of core values and responsible strategies for protecting the privacy interests of participants and properly managing personal information, including health information and genomic data, *available at*: https://www.nih.gov/sites/default/files/research-training/initiatives/pmi/privacy-trust-principles.pdf.

[3] The *PMI Security Principles* document is a set of security policy principles and a framework for protecting the security of participants' data and resources in an appropriate and ethical manner, *available at*: https://www.nih.gov/sites/default/files/research-training/initiatives/pmi/security-principles-framework.pdf.

[4] The NIST Standards are for informational purposes only, as they may reflect current best practices in information technology and are not required for compliance with the HIPAA Rules.

[5] For more information about the HIPAA Privacy, Security, and Breach Notification Rules, please visit the HHS Office for Civil Rights (OCR) Health Information Privacy website, *available at*: https://www.hhs.gov/hipaa/index.html.

[6] The considerations provided in this document are for informational purposes only as they reflect best practices in privacy and security for health information technology and do not guarantee compliance with Federal, state, or local laws. This document is not intended to be an exhaustive or definitive source of safeguarding health information privacy and security risks. This document is not intended to serve as legal advice or as recommendations based on a provider or professional's specific circumstances. We encourage organizations to seek expert advice when evaluating the use of these considerations.

2. Enable technology to provide for and respect individuals' choices and/or preferences about the specific types of health information (e.g., medication lists, allergies) shared with the third-party.
3. Provide methods for individuals to revoke permissions for sharing health information about them in a manner that is clear and easily accessible.
4. Develop organizational privacy policies that are consistent with the PMI Privacy Principles and adequately address privacy risks.

KEY SECURITY CONSIDERATIONS

The following are key areas for security consideration when implementing APIs in healthcare:

1. Use Transport Layer Security (TLS) Version 1.2[7] or higher with strong cipher suites (such as the Advanced Encryption Standard [AES] or higher) to protect health information in transit via the API from the EHR to the third-party.
2. Ensure that the API cannot be manipulated to unintentionally expose health information or system vulnerability information.
3. Develop technical and administrative policies to ensure verification of the identity of users and contributors, prior to granting credentials for access to or contribution of health information.
4. Develop technical and administrative policies that describe how to issue credentials to individuals that will permit them to access health information about themselves.
5. Consider implementing risk-based authentication controls that flow from the organization's security risk assessment, and are commensurate with the type of data, level of sensitivity of the information, and user type.
6. Develop systems with technical authorization controls flexible enough to support individual privacy preferences that are capable of limiting API access, use, or disclosure based on what is necessary to satisfy a particular purpose or carry out a function.
7. Evaluate any service provider's infrastructure, security practices, and technical capabilities for hosting implementations of APIs and apps that store and access health information.
8. Implement data integrity protection controls that detect when unauthorized alterations are made to health information made accessible through the API.
9. Ensure that EHR patient portals that interact with the API are secure and protected against known vulnerabilities that attackers could exploit.
10. Develop organizational security policies that are consistent with the PMI Security Principles and adequately address security risks.

While there is a perception in the healthcare community that APIs are less secure than other components of an IT system (e.g., an EHR system),[8] the use of APIs to share information is prevalent in the financial and travel industries. In turn, similar to the use of APIs in other industries, as long as healthcare APIs are implemented with appropriate privacy and security safeguards in place, APIs can add value to individual-directed sharing of health information. As adoption of API use in healthcare becomes more widespread, this document can serve as a resource for specific privacy and security considerations for health information technology (health IT) implementers.

[7] https://tools.ietf.org/html/rfc5246
[8] https://www.healthit.gov/hitac/sites/faca/files/HITJC_APITF_Recommendations.pdf

INTRODUCTION

The Precision Medicine Initiative, launched in 2015, is a nationwide initiative to move away from the "one-size-fits-all" approach to healthcare delivery and instead, tailor treatment and prevention strategies to people's unique characteristics, including environment, lifestyle, and biology. The *All of Us* Research Program (*All of Us*),[9] which is led by the National Institutes of Health (NIH) and is a key component of the PMI, aims to gather health data from one million or more people in the United States and make those data available to researchers with the goal of accelerating the pace of research and improving health.

Sync for Science (S4S) is a collaboration among researchers, electronic health record (EHR) developers, healthcare provider organizations, the Office of the National Coordinator for Health Information Technology (ONC), and NIH.[10] S4S coordinates pilot organizations (EHR developers and associated healthcare providers) so that participants[11] have the ability to share EHR-based clinical-health data[12] with a research program of their choice, such as *All of Us*, via a read-only[13] API.

An API is technology that allows one software app to programmatically access the services provided by another software app.[14] For example, third-party online travel agents use APIs, provided by each individual airline, to access flight scheduling data and aggregate information for consumers to find an optimal flight. Similarly, the healthcare industry can use APIs. In particular, the S4S API enables participants (e.g., patients) who enroll in the research initiative to make a choice to share health information related to or about them directly with *All of Us* by submitting a signed, electronic, HIPAA right of access request for their healthcare provider to transmit health information about them from an EHR system to a designated third-party, in this case, the *All of Us* app.

While health data may have its own distinct characteristics (e.g., format, legal and regulatory data handling requirements), the technical processes by which health data can be used and accessed via APIs mirrors other commercial implementations. Use of APIs in healthcare has the potential to provide seamless electronic transmission of information between health IT systems and third-party apps but may also introduce risks and vulnerabilities.[15] However, in 2016, the Joint API Task Force of the Health IT Policy and Standards Federal Advisory Committees (FACA) recommended to ONC that the benefits of API-enabled health information exchange outweigh potential security risks. The Task Force noted that APIs may be difficult to implement with respect to protecting health information, but, if appropriately

[9] http://allofus.nih.gov/
[10] https://www.healthit.gov/buzz-blog/health-innovation/nih-and-onc-launch-the-sync-for-science-pilot/
[11] *All of Us* direct volunteers and healthcare provider organization-based volunteers. https://www.nih.gov/research-training/allofus-research-program
[12] Health information sharing functionality supports requirements for enabling individuals' HIPAA Privacy Rule right to access health information about them. https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/access/
[13] Read-only implies that the API allows access to retrieve information from the data source but does not allow modification or addition of information to the data source via the API.
[14] ONC has produced an educational resource about APIs in healthcare, *available at*: https://www.healthit.gov/api-education-module/story_html5.html
[15] Vulnerabilities are flaws or weaknesses in a system that potentially exposes health information. Risk is the potential for loss, damage or destruction of an asset as a result of threat exploiting vulnerabilities.

managed, can result in benefits for patients and providers. Testimony given to the Task Force stated that there are two primary categories of API vulnerabilities:[16]

- API vulnerabilities due to imperfect or outdated internet, web, and API security specifications
- API vulnerabilities due to human oversight. This includes ignoring certain security best practices or poorly designed APIs that result in unintended functionality

The testimony noted that, taken together, both human and technical challenges can make it possible for even the biggest internet companies to publish improperly secured APIs.[17]

Overall, the Task Force recommended that read-only APIs could provide better health information security than ad-hoc interfaces or proprietary integration technologies so long as they are implemented with appropriate privacy and security safeguards in place. While access to health information via APIs requires further privacy and security considerations, APIs can add value to individual-directed access when incorporated with existing standards, infrastructure, and identity-proofing processes.

This section describes how implementers such as healthcare providers and EHR developers enable an individual to use the S4S API to make a HIPAA right of access request to share health information about them with a designated third-party app. Moreover, to help ensure that appropriate privacy and security safeguards are in place to enable this individual-directed exchange of health information, ONC, with support from NIH, contracted with an organization, independent of the S4S project, to conduct testing and assessment on volunteer healthcare providers and EHR developers participating in S4S.

S4S Technology & Functionality

The S4S API is based on the Health Level 7 (HL7®) Fast Healthcare Interoperability Resources (FHIR®) Draft Standard for Trial Use 2 (DSTU2)[18] and OAuth 2.0[19] framework. The S4S API requires implementers to use the Substitutable Medical Apps, Reusable Technology (SMART) App Authorization Guide,[20] which describes how app developers can access FHIR resources using OAuth 2.0.[21] OAuth 2.0 provides electronic access control mechanisms based on rules set to enforce a healthcare provider's organizational security policy. The S4S API uses OAuth 2.0 to allow a designated third-party app (e.g., *All of Us*) to have electronic, read-only access to all or a portion of health information about an individual, made available through a healthcare provider's EHR patient portal, via the individual's existing authentication credentials (e.g., username and password).[22] The S4S API enables a healthcare provider to share a subset[23] of the Certified EHR Technology (CEHRT) Common Clinical Data Set (CCDS) data elements,[24] consistent with the Argonaut Project FHIR profile.[25] The S4S API was developed with the intention that it can be used for other third-party apps, including those for medical research.

---

[16] https://www.healthit.gov/hitac/sites/faca/files/HITJC_APITF_Recommendations.pdf
[17] https://www.healthit.gov/hitac/sites/faca/files/APITF_Testimony_DavidBerlind_2016-01-26.pdf
[18] https://www.hl7.org/fhir/DSTU2
[19] https://tools.ietf.org/html/rfc6749
[20] http://docs.smarthealthit.org/authorization/
[21] http://docs.smarthealthit.org/authorization/
[22] Automatic electronic access of health information through the API is only available after the individual submits an electronic HIPAA access request to her healthcare provider to transmit health information related to her from the healthcare provider's EHR to the *All of Us* app.
[23] http://syncfor.science/api-calls/
[24] https://www.healthit.gov/sites/default/files/2015Ed_CCG_CCDS.pdf

S4S User Story

S4S API Functionality

The S4S API enables individuals to request the transfer of health information about them, from their healthcare providers' EHR, to the *All of Us* Data and Research Center (DRC) through a third-party app (e.g., *All of Us* app).

The following user story describes the expected workflow for an individual[26] using the *All of Us* app to facilitate health information sharing from her healthcare provider's EHR to *All of Us*.[27]

An individual decides she wants to voluntarily enroll in *All of Us* and share health information maintained in her healthcare provider's EHR.[28] Once enrolled in *All of Us*, she accesses the *All of Us* app, either via a web browser or mobile application.[29] In the app, the individual or an authorized user[30] selects her specific healthcare provider, whose EHR contains health information about her. The app then directs the individual to access that healthcare provider's EHR patient portal[31] and log in using her credentials. The patient portal displays the healthcare provider's electronic form, which allows the individual to select specific subsets of health information to share, and confirm the request to transfer health information to *All of Us* by selecting the "approve" button. As a result of the individual clicking "approve," the *All of Us* app is allowed read-only access to the health information maintained in the provider's EHR using the S4S API.[32] The individual is redirected back to the app and, by utilizing the S4S

---

[25] http://argonautwiki.hl7.org/index.php?title=Main_Page

[26] The use of the term patient and/or individual in the user story and elsewhere in the document parallels the *All of Us* use of the term "participant." S4S participants are individuals who are patients of healthcare providers who have health information about them that they want to share with *All of Us* and have chosen to participate in the program.

[27] Transmission of health information between the *All of Us* Participant Technologies Center (PTC), identified as the "app," and the DRC repository, is to be assessed by other independent third-party security experts and does not involve the S4S API. The PTC is a third-party entity contracted to develop, test, maintain, and upgrade (as needed) the *All of Us* app.

[28] Identity proofing and credentialing of *All of Us* participants occurs during the enrollment process at the healthcare provider level, and the *All of Us* app relies on the healthcare provider's patient portal to accurately and electronically convey patient identity for purposes of information sharing through the S4S API. https://allofus.nih.gov/sites/default/files/allofus-initialprotocol-v1_0.pdf

[29] *All of Us* Direct Volunteer participants who have enrolled at one of these pilot sites will be able to sign into their healthcare provider's patient portal using the S4S workflow and authorize sharing health information about them with the program. https://allofus.nih.gov/sites/default/files/allofus-initialprotocol-v1_0.pdf

[30] An individual's personal representative is generally, a person with authority under state law to make healthcare decisions for the individual. For more information about an individual's personal representative's access rights under the HIPAA Privacy Rule, please view the HHS OCR Access Guidance *available at*: https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/access/index.html.

[31] A patient portal is a secure website that gives individuals electronic access to personal health information (e.g., protected health information contained in a provider's EHR system) from anywhere with an internet connection. https://www.healthit.gov/providers-professionals/faqs/what-patient-portal

[32] For purposes of this document, it is assumed that an individual's request to transfer health information will be fulfilled automatically. Any reason to deny the transfer (e.g. HIPAA limited exceptions) is outside the scope of this document. Depending on the healthcare provider's implementation of the S4S API, an individual may be able to limit the categories of health information shared with *All of Us* (e.g. demographics, medications, vital signs), as well

API, the healthcare provider transmits the health information to *All of Us*.[33] Once the individual has agreed to share health information with a third-party, the third-party will be able to access the health information until one year has passed, or the individual decides to revoke access/permission to share, whichever occurs first.

Figure 1 depicts a technical view of the S4S user story from the perspective of a user (*All of Us* participant).[34]

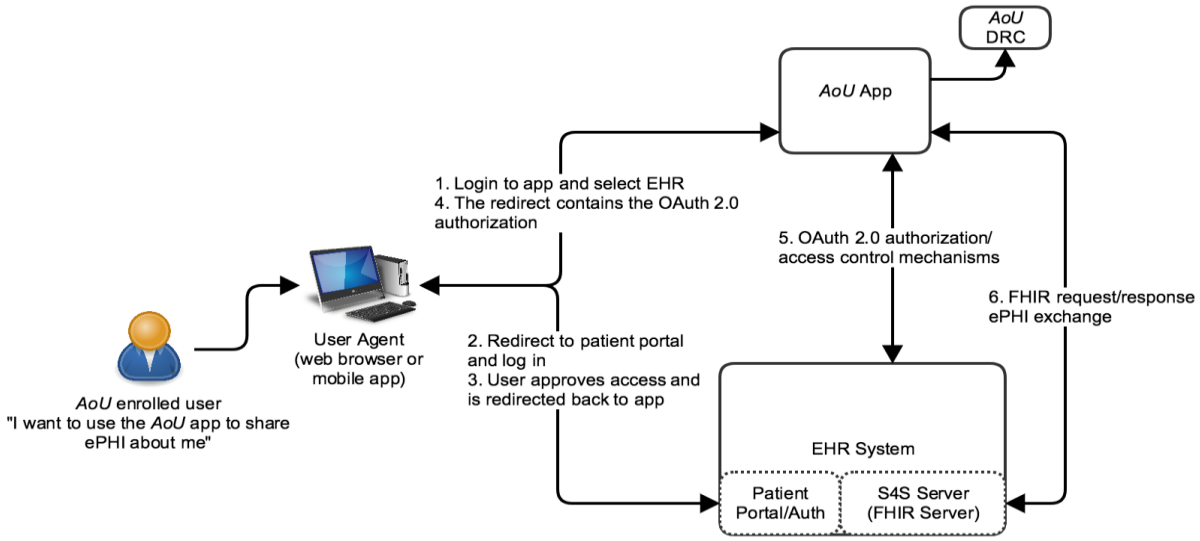

Figure 1: S4S Technical User Story Diagram

Sync for Science API Privacy and Security (S4SPS)

ONC led the Sync for Science API Privacy and Security project (S4SPS) to assess whether a voluntary subset of S4S pilot organizations applied appropriate privacy and security principles for their implementations of the S4S API. S4SPS reviewed the implementations of the S4S API across various users, system processes, policies, and technology components for health information confidentiality, integrity, and availability (CIA), in accordance with applicable principles from the PMI Privacy Principles[35] and PMI Security Principles.[36] Special publications from NIST were also used as a basis for assessment criteria of the participating S4S pilot organizations. Entities covered by HIPAA must implement appropriate privacy protections and data security safeguards in their environment, and in particularly,

---

as a date when she would like the healthcare provider to stop sharing health information through the S4S API with *All of Us*.

[33] For purposes of this use case, sharing health information with the *All of Us* app is equivalent to sharing health information with a designated third-party researcher. For more information about honoring an individual's access request to send health information to a designated third-party, please view the HHS OCR Access Guidance *available at*: https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/access/index.html.

[34] S4SPS assumes the "standalone launch" workflow, in which the *All of Us* app launches outside of an EHR session, is in use, as opposed to the "EHR launch" workflow, in which the *All of Us* app launches from within an existing EHR session. http://syncfor.science/api-calls/

[35] https://www.nih.gov/sites/default/files/research-training/initiatives/pmi/privacy-trust-principles.pdf

[36] https://www.nih.gov/sites/default/files/research-training/initiatives/pmi/security-principles-framework.pdf

comply with the HIPAA Privacy and Security Rules.[37] The PMI Privacy and Security Principles are consistent with HIPAA and can help bolster an entity's privacy and security posture.

The following KEY PRIVACY CONSIDERATIONS and **Error! Reference source not found.**,[38] gleaned from S4SPS testing and assessment results, describe tips for health IT implementers to consult for future deployment of APIs in healthcare.

KEY PRIVACY CONSIDERATIONS

These privacy considerations for healthcare API implementations reflect key areas with respect to protecting health information and are priorities for future healthcare APIs.[39]

Enabling an Individual's HIPAA Right of Access

Key Consideration
Ensure that any electronic access request interface (e.g., an electronic form within a patient portal) provides individuals with an opportunity to approve the electronic transmission of health information in accordance with applicable legal requirements that enable such a request (e.g., HIPAA right of access).

Background – To enhance transparency, the PMI Privacy Principles suggest communicating to participants clearly and conspicuously concerning: how, when, and what information will be collected and stored; to explain how health information about them will be used, accessed, and shared; and the goals, potential benefits, and risks of participation, including risk of inappropriate use or compromise of the information about participants. Any language provided to an individual that describes the legal mechanism for enabling sharing of health information should be similarly transparent. For example, under the HIPAA right of access, the individual's request to direct health information to a third-party must be in writing, signed by the individual, and clearly identify the designated person or entity, of the individual's choice, on where to send the health information. A covered entity, including a healthcare provider, may accept an electronic copy of a signed request (e.g., PDF), as well as an electronically executed request (e.g., via a secure web-based patient portal) that includes an electronic signature.[40]

Tips for Implementers

- Capture a valid electronic signature by developing an interface that allows individuals to click a graphical button that reads, "Approve," which follows a clear written description of the health

[37] For more information on how to determine whether or not you are HIPAA covered entity or business associate, see the HHS OCR Guidance, *available at*: http://www.hhs.gov/hipaa/for-professionals/covered-entities/.

[38] The considerations provided in this document are for informational purposes only as they reflect current best practices in privacy and security for health information technology and do not guarantee compliance with Federal, state, or local laws. This document is not intended to be an exhaustive or definitive source of safeguarding health information privacy and security risks. This document is not intended to serve as legal advice or as recommendations based on a provider or professional's specific circumstances. We encourage organizations to seek expert advice when evaluating the use of these considerations.

[39] In addition to the considerations listed, privacy requirements under applicable laws and regulations may apply to an organization.

[40] https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/access/index.html

information, about the identified individual, to be shared with a third-party, and a clear identification of the third-party designated to receive the health information.[41]

- Ensure that the method of approval does not imply that the HIPAA right of access request is a "HIPAA Authorization," which is a separate method of sharing health information about an individual.[42]
- In accordance with the PMI Privacy Principle of transparency, the approval text should make clear to the individual that once the third-party receives the transmitted health information, the healthcare provider is not responsible for what happens to the information maintained by the third-party.

Scope & Granularity of Patient Choice

Key Consideration

Enable technology to provide for and respect individuals' choices and/or preferences about the specific types of health information (e.g., medication lists, allergies) shared with the third-party.

Background – According to the PMI Principle of respecting patient preferences, the *All of Us* should promote individual autonomy and trust through a dynamic and ongoing consent and information sharing process. This process should enable individuals to engage actively in an informed and voluntary manner, and to reevaluate their own preferences as data sharing, use requirements, and technology evolve.[43]

Tips for Implementers

- Provide individuals with specific options about what health information is shared with the third-party app instead of allowing all-or-nothing access to health information. These options could enable an individual to select specific categories of health information about them (e.g., allergies, smoking status, demographic information as defined by the CCDS) to share with the third-party. Where possible, API developers should enable these types of granular sharing of health information as supported by the SMART App Authorization Guide. [44]

---

[41] Implementers can reference the Electronic Signatures in Global and National Commerce (ESIGN) Act when developing the mechanism for capturing electronic signatures. https://www.gpo.gov/fdsys/pkg/PLAW-106publ229/html/PLAW-106publ229.htm

[42] For more information about the difference between an individual's HIPAA access right and a HIPAA authorization, please view the HHS OCR Access Guidance, *available at*: https://www.hhs.gov/hipaa/for-professionals/faq/2041/why-depend-on-the-individuals-right/index.html

[43] https://allofus.nih.gov/sites/default/files/privacy-trust-principles.pdf

[44] The most granular access scope that the SMART App Authorization Guide currently permits is a FHIR resource. Meanwhile, the S4S technical specification handles sharing of data elements from the CCDS, which are more granular than a FHIR resource. The standards development organization of SMART on FHIR standards should consider developing scopes that are granular enough to reflect each of the CCDS data elements in the future. Without a mechanism in a patient portal to limit the amount and type of health information that can be shared from the EHR, health information that an individual does not desire to share may be available to the third-party app. For API developers enabling fine-grained health information access, the SMART App Authorization Guide provides examples which can be used to define specific OAuth 2.0 access scopes to indicate which FHIR resource types an individual or authorized user is allowed to access when requesting individual health information from an EHR system.

Methods for Revoking Sharing Permissions

Key Consideration
Provide methods for individuals to revoke permissions for sharing health information about them in a manner that is clear and easily accessible.

Background – In accordance with the PMI Privacy Principle to respect patient preferences, healthcare API implementations should allow individuals to withdraw their permission (e.g., access request to share health information with a third-party) for future use and data sharing at any time and for any reason, with the understanding that permission for use of data included in aggregate data sets or in past studies and studies that have already begun cannot be withdrawn.

Tips for Implementers

- Include built-in functionality within the EHR patient portal to allow individuals a way to view and manage the third-party apps that have access to health information about them. This will help ease potential uncertainty caused by the varying nature in which different third-party apps might present individuals with an opportunity to revoke permission to share health information. This functionality could allow the individual to revoke the HIPAA access request to share health information with a third-party, at any point.[45]

Development of Organizational Privacy Policies

Key Consideration
Develop organizational privacy policies that are consistent with the PMI Privacy Principles and adequately address privacy risks.

Those who access and use health IT should protect health information that they maintain for transmission through an API. When developing an organizational privacy policy, organizations can use NIST 800-53 privacy controls to serve as a data privacy roadmap. The following table lists a subset of the privacy control families that are most relevant to implementing an API in healthcare and the corresponding key considerations.

NIST Privacy Control Family	Organizational Privacy Policy Key Considerations for API Implementation
Authority and Purpose (AP)	Prior to API implementation, document why the API collects health information and ensure that necessary agreements are in place with the healthcare provider before the EHR developer begins handling health information.

[45] The Joint API Task Force recommended that patient portals should: Show individuals a list of all active app authorizations in the portal; Include the ability for the individual to revoke any app authorization; Show individuals a list of which apps have accessed health information about them via the API (including relevant details). https://www.healthit.gov/hitac/sites/faca/files/HITJC_APITF_Recommendations.pdf

NIST Privacy Control Family	Organizational Privacy Policy Key Considerations for API Implementation
Accountability, Audit, and Risk Management (AR)	Ensure effective administrative, technical, and physical controls for governance, monitoring, risk management, and assessment to demonstrate adherence to applicable privacy protection requirements while minimizing overall privacy risk when implementing the API.
Data Quality and Integrity (DI)	Specify delegation of responsibilities between healthcare providers and EHR developers to ensure data privacy and integrity; including periodic auditing to ensure health information exposed to an API is continuously accurate, relevant, timely, and complete.
Data Minimization and Retention (DM)	Specify that health information collection through an API is strictly limited only to that which is necessary to accomplish the original purpose of the collection or use and is retained for only as long as it is needed.
Individual Participation and Redress (IP)	Specify the ability for an individual to be able to opt-in and out of API health information sharing.
Security (SE)	Specify delegation of responsibilities between healthcare providers and EHR developers to define how privacy incident monitoring and response should be handled in the event of an incident.
Transparency (TR)	Give individuals clear notice regarding why health information about them is collected, how it is used, and to whom it might be disclosed via a notice of privacy practices.
Use Limitation (UL)	Ensure health information is used in a manner that is consistent with representations made in a notice of privacy practices.

Table 1: Key Privacy Control Considerations for Organizational Privacy Policy – API Implementation

KEY SECURITY CONSIDERATIONS

These security considerations for healthcare API implementation reflect key areas with respect to safeguarding health information and are priorities for future healthcare APIs.[46]

Encryption of Data in Transit

Key Consideration
Use Transport Layer Security (TLS) Version 1.2 or higher with strong cipher suites (such as the Advanced Encryption Standard (AES) or higher) to protect health information in transit via the API from the EHR to the third-party.

[46] In addition to the considerations listed, security requirements under applicable laws and regulations may apply to an organization.

Background – The PMI Security Principles seek to ensure the protection and security of data in-transit through the use of strong encryption. Strong encryption methods establish a secure connection and prevent leakage of any health information or metadata in-transit between the app and servers.[47] The SMART App Authorization Guide and NIST SP 800-52[48] recommend the use of TLSv1.2,[49] which contains a stronger set of ciphers[50] that make it less likely for an attacker to exploit a communications channel.

While exploiting weaknesses in any version of Secure Sockets Layer (SSL) and TLS requires a high skill level for a would-be attacker, the public body of knowledge regarding exploits for earlier versions of SSL/TLS is far more expansive. As such, using up-to-date versions of encryption methodologies helps to minimize the risks to sharing health information. There are no fixes or patches that can adequately repair SSL or early TLS. Therefore, it is important that organizations upgrade to TLSv1.2 or higher as soon as possible, and disable any fallback to both SSL and early TLS.

Tips for Implementers

- Ensure the secure usage of APIs and protect the confidentiality of data transmitted (e.g., health information or associated metadata) between a third-party app and any servers involved in the transmission of health information (e.g., API server, patient portal/authorization server, or token server) by having all electronic communication involving health information use an up-to-date version of TLS.
    - Use TLSv1.2 with strong cipher suites (such as AES or higher) for exchanging data.
    - Disable support of lower versions of TLS,[51] which contain weak cipher suites, to protect against attacks and security vulnerabilities that are known to exist and could lead to interception or modification of transmitted data.

Input Validation of API Calls

Key Consideration

Ensure that the API cannot be manipulated to unintentionally expose health information or system vulnerability information.

Background – The PMI Security Principles assert that health information should be protected by cybersecurity controls and ensure data integrity. These cybersecurity controls will help to prevent an attacker from being able to exploit vulnerabilities and modify health information stored within the system using techniques such as fuzzing,[52] invalid input attacks,[53] and injection attacks.[54,]

[47] http://docs.smarthealthit.org/authorization/best-practices/
[48] http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf
[49] TLS is a cryptographic protocol used to establish a secure communications channel between two systems. It is used to authenticate one or both systems, and protect the confidentiality and integrity of information that passes between systems through encryption, which converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext. https://tools.ietf.org/html/rfc5246
[50] Cipher suites are a combination of algorithms used for negotiating the settings of a TLS connection. http://csrc.nist.gov/publications/fips/fips140-2/fips1402annexa.pdf
[51] Many serious vulnerabilities exist in TLS's predecessor, SSL, and early TLS protocols (TLSv1.1 and earlier) that, left unaddressed, put organizations at risk of experiencing a security incident.
[52] Fuzzing is an automated method for testing APIs by inputting randomized data into API calls to create a system error or message that could inadvertently expose information. https://www.owasp.org/index.php/Fuzzing

Tips for Implementers

- Use cybersecurity software tools, particularly validation and sanitization libraries or frameworks, to ensure that incoming API requests are formed and encoded in a manner that removes potentially unsafe elements, before they are executed on the EHR. For example, if a third-party app makes an API call requesting health information related to an individual's medication from the EHR, but the API call from the third-party includes extra or irrelevant data, the validation and sanitization libraries can detect the extraneous data and ensure that no malicious inputs are entered into the EHR.
- Verify that the API implementation does not reveal or display detailed error information that could make it easier for an attacker to craft a targeted attack and potentially exploit vulnerabilities in the system. For example, an API might reveal this detailed error information when returning a response to a third-party app if the API is implemented in a way that supports verbose error messages. Disable any messages that may have been used for debugging or error-trapping purposes in a development environment to limit the exposure of information that may make an EHR vulnerable to attack.

Access Controls – Protecting Against Unauthorized Access

The following subsections: Access Controls – Identity Proofing and Access Controls – Credentialing represent aspects of health information security that occur at the healthcare provider level, prior to an individual gaining access to a healthcare provider's patient portal or a third-party app (e.g., *All of Us* research app).[55]

Access Controls – Identity Proofing

Key Consideration

Develop technical and administrative policies to ensure verification of the identity of users and contributors, prior to granting credentials for access to or contribution of health information.

Background – Under the PMI Security Principles, health IT developers, manufacturers, clinicians and others who use EHRs should develop a policy for verifying the identity of users and contributors (e.g., individuals and healthcare provider staff), prior to granting credentials (e.g., username and password) for access to or contribution of data for third-party (e.g., research) purposes.

Strong identity-proofing policies are essential for the privacy and security of health information because third-party apps such as the *All of Us* app, which often utilize APIs to access health information, rely on credential-issuing parties (e.g., healthcare providers) to properly identity proof the individual and

[53] Invalid input attacks exploit a targeted system by sending structured inputs that are contrary to what is expected. https://www.owasp.org/index.php/Input_Validation_Cheat_Sheet.

[54] Injection attacks insert code into a system (e.g., the EHR via the API) to prompt the execution or interpretation of the inserted code with a malicious intent, for example, through SQL injections, in which malformed SQL queries are inserted into an app. https://www.owasp.org/index.php/Code_Injection. See also https://www.owasp.org/index.php/REST_Security_Cheat_Sheet

[55] Testing identity-proofing and credentialing policies were out of scope of the S4SPS assessment, as identity proofing and credentialing are conducted during the *All of Us* enrollment phase, prior to the individual gaining access to the healthcare provider patient portal and the *All of Us* research app.

establish that a person is who he or she claims to be. Identity proofing is the first step before issuing an individual credentials to access an EHR patient portal.

Tips for Implementers

- Ensure that the level of assurance[56] for identity proofing reflects the appropriate risk, given the issued party's exposure to health information. Strong organizational identity-proofing policies help prevent the issuance of credentials to malicious actors, therefore, blocking them from gaining unauthorized access to health information. Implementers can look to NIST standards[57] or other sources for identity proofing best practices to use when granting access to individuals.

Access Controls – Credentialing

Key Consideration

Develop technical and administrative policies that describe how to issue credentials to individuals that will permit them to access health information about themselves.

Background – The PMI Security Principles state that data quality and integrity should be maintained at all stages—collection, maintenance, use, and dissemination. Credentials, issued after proper identity proofing, are what individuals actually use, or say they are, in order to gain access to health information about them maintained in their healthcare provider's systems.[58] Examples of credentials include usernames and passwords used to access an individual account on a patient portal.

Tips for Implementers

- Require users to create login credentials that are sufficiently complex so that they are not easily discoverable or predictable by malicious actors.
- Hash passwords using a one-way algorithm, such as the Secure Hash Algorithm-2 (SHA-2) family, and store passwords securely to prevent an attacker from being able to decrypt the passwords and use them to gain access to an individual's account through the patient portal, in the event that the passwords are stolen.[59]

Verification of Access Controls

The following subsections:

Verification of Access Controls – Authentication and Verification of Access Controls – Authorization (OAuth 2.0) represent how an individual's identity-proofed credentials should be utilized to verify an individual is who they claim to be (authentication) and that the individual has been granted access to the information they are attempting to access (authorization).

---

[56] Level of assurance, as defined by the International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 29115 standard, describes the degree of confidence in the processes leading up to and including an authentication. It provides assurance that the entity claiming a particular identity is the entity to which that identity was assigned.

[57] http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63a.pdf

[58] http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf

[59] https://pages.nist.gov/800-63-3/sp800-63b.html

Verification of Access Controls – Authentication

Key Consideration

Consider implementing risk-based authentication controls that flow from the organization's security risk assessment, and are commensurate with the type of data, level of sensitivity of the information, and user type.

Background – Authentication is the process of verifying the identity of an individual using credentials to access a system, for example, verifying the username and password an individual enters when logging in to a patient portal.[60]

Tips for Implementers

- Consider using the OpenID Connect[61] authentication protocol, in combination with OAuth 2.0 so that third-party apps can verify the identity of individuals who need to access health information about them through secure APIs, during authentication with a patient portal in an interoperable and secure manner. Using OpenID Connect should help provide assurance that the users (e.g., patients) of the apps are who they claim to be and that the users are authorized to access the health information that is shared with the apps (e.g., an individual using a given set of credentials is in-fact the individual to which those credentials have been assigned).
- Support strong multi-factor authentication rather than single-factor authentication when users need to access a patient portal.[62]

Verification of Access Controls – Authorization (OAuth 2.0)

Key Consideration

Develop systems with technical authorization controls flexible enough to support individual privacy preferences that are capable of limiting API access, use, or disclosure based on what is necessary to satisfy a particular purpose or carry out a function.

Background – According to the PMI Security Principles, authorization controls should be granular enough to support participant choice and should limit access, use, or disclosure based on what is necessary to satisfy a particular purpose or carry out a function. Authorization (or access control) is the process of verifying that an authenticated person has the proper permissions to access the system and the associated information within the system. The OAuth 2.0 framework includes support for different workflows that health IT users or developers can implement to authorize a third-party app to access resources intended for the specific user of the app.[63] As part of OAuth 2.0, health IT developers, manufacturers, owners, and other users must implement an authorization server that is responsible for

[60] http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf
[61] OpenID Connect is an interoperable identity layer on top of the OAuth 2.0 protocol. It allows third-party apps to verify the identity of the end-user (e.g., patient) based on the authentication performed by a patient portal. http://openid.net/connect/
[62] https://pages.nist.gov/800-63-3/sp800-63b.html - sec5
[63] https://tools.ietf.org/html/rfc6749

handling authorization for apps wishing to gain access to protected information such as health information.

Tips for Implementers

- Use the technical OAuth 2.0 framework, as specified by the SMART App Authorization Guide,[64] to enable a third-party app to securely obtain authorized access to health information from an EHR system (e.g., patient portal) through an API for a specified duration and prevent unauthorized parties from accessing the health information.
- Implement the "authorization code" grant type workflow recommended by the SMART App Authorization Guide because it enables an individual to authenticate directly to an authorization server without sharing credentials with the app and allows the server to transmit access and refresh tokens directly to the app without being exposed to intermediaries, such as a web browser.[65]
- Support required parameters in authorization requests prior to obtaining authorization codes that an app will use to exchange for an access token, in accordance with the SMART App Authorization Guide.
- Support required parameters when making requests for access tokens that must be presented when making API requests to access health information.
- Support additional request parameters recommended by the SMART App Authorization Guide, short lifetimes for authorization codes (e.g., one minute) and access tokens (e.g., one hour), and required authentication of third-party apps when obtaining access tokens, to prevent against and reduce the risk of malicious attacks against their API implementations.

Service Provider Security

Key Consideration

Evaluate any service provider's infrastructure, security practices, and technical capabilities for hosting implementations of APIs and apps that store and access health information.

Background – As stated in the PMI Security Principles, when organizations employ subcontractors, third parties, or vendors (including hosted, cloud, or app service providers) to create, receive, maintain or transmit health information, those organizations should obtain the necessary assurances that the service provider will appropriately safeguard health information, consistent with the organization's security policy.

Tips for Implementers

- Consider the overall infrastructure and other software the API interfaces with, within the technical environment, to help mitigate any potential vulnerabilities. Health IT users should understand how the service provider protects and secures the infrastructure through the design and deployment of security and operation practices, techniques, and capabilities because the technical environment that hosts the API could be the source of vulnerabilities that could leave

[64] http://docs.smarthealthit.org/authorization/
[65] https://tools.ietf.org/html/rfc6749 - section-4.1

an API vulnerable to attacks and health information leakage.[66] As changes are made to other components of the EHR system, the healthcare provider or health IT developer should perform security testing on the API and the components of the EHR system the API interfaces with to ensure that new functionalities do not introduce new vulnerabilities to the API.

- Execute and maintain a service level agreement (SLA) with the service provider that defines uptime and performance requirements regarding the availability of the infrastructure that hosts an implementation of the API, as well as the availability of the health information the service provider maintains or transmits via an API.[67]

Data Integrity Protection

Key Consideration

Implement data (e.g., health information) integrity protection controls that detect when unauthorized alterations are made to health information made accessible through the API.

Background – The PMI Security Principles recommend that health IT developers and others that support these systems implement integrity protection controls that detect when unauthorized alterations have been made to health information. Data integrity is the assurance that health information has not been changed, destroyed, or lost in an unauthorized or accidental manner while in storage, during processing, and while in transit. In addition to monitoring unauthorized changes to health information, health IT developers can protect the integrity of health information by preventing the disclosure of information about API implementations that attackers could use to craft targeted attacks with the intention to modify health information.

Tips for Implementers

- Use data integrity monitoring software to help detect unauthorized changes to health information as it flows through an API and remediate any unauthorized changes to the health information, in combination with encryption.
- Review system and information integrity (SI) policies and procedures periodically for alignment with the NIST 800-53 SI security control family.[68]
- Ensure that Hypertext Transfer Protocol (HTTP) headers of a web server and API error messages or faults do not disclose detailed information about the underlying web server that could be the source of potential exploitation.
- Consider using additional methods of security for an API to help authenticate where Domain Name System (DNS) responses are coming from and ensure that they are valid. For example, the use of Domain Name System Security Extensions (DNSSEC), a suite of extensions that add security to the DNS protocol, can ensure that domains associated with API endpoints that transmit health information or information required for API access are secure.[69]

---

[66] APIs could be hosted in many different ways such as on systems maintained on the physical premise of the developer or healthcare provider site, or in the virtual cloud with a service provider like Amazon Web Services or Microsoft Azure, or even a hybrid of on premise and the cloud.

[67] See, e.g., https://www.hhs.gov/hipaa/for-professionals/special-topics/cloud-computing/index.html

[68] https://nvd.nist.gov/800-53/Rev4/control/SI-1

[69] Specifically, DNSSEC provides origin authority, data integrity, and authenticated denial of existence. With DNSSEC, the DNS protocol is much less susceptible to certain types of attacks, particularly DNS spoofing attacks.

Patient Portal Security

Key Consideration

Ensure that EHR patient portals that interact with the API are secure and protected against known vulnerabilities that attackers could exploit.

Background – The PMI Security Principles assert that health IT manufacturers, developers, and providers who oversee EHR systems should use risk-management strategies, tools, and techniques to inform and prioritize decisions regarding the protection of health information, including data in electronic and physical resources within its environment as well as at the point of initial collection.

Patient portals provide individuals with access to health information about them and can also include the functionality for an individual to share health information with a third-party through the use of an API. For example, a patient portal is the user interface where individuals choose to share health information with a third-party app, prior to the healthcare provider transmitting the health information to the app via an API. Patient portals are endpoints of an EHR system that interact directly with APIs. As such, they present opportunities for would-be attackers to input untrusted data or malicious code into an EHR system, if left unsecure.

Securing the patient portal with security-encoding libraries limits the risk of use of untrusted data on patient portal pages and properly encodes any untrusted data to reduce the number of vulnerabilities that an attacker could exploit, preventing unauthorized access to health information within the patient portal, either directly when logged-in, or indirectly, via an API.[70]

An example of a vulnerability that an attacker could exploit on a patient portal is a cross-site scripting (XSS) attack. XSS attacks can occur when an attacker directs a victim to a malicious URL, a URL containing a malicious script, or a victim visits a website that contains a malicious script injected by an attacker. The malicious script would run in the user agent, which is software that acts on behalf of the user, potentially transmitting sensitive information somewhere other than the intended destination.

Tips for Implementers

- Protect unauthorized access to health information by preventing successful XSS attacks on patient portal pages by using security-encoding libraries to limit the use of untrusted data on patient portal pages and to properly encode any untrusted data.

Development of Organizational Security Policies

Key Consideration

Develop organizational security policies that are consistent with the PMI Security Principles and adequately address identified security risks.

[70] https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

The PMI Security Principles use the NIST Framework for Improving Critical Infrastructure Cybersecurity (the Cybersecurity Framework) to define a set of activities, outcomes, and references that, when followed, enable five simultaneous and continuous functions — *Identify, Protect, Detect, Respond,* and *Recover* — to assess cybersecurity and data security performance, as well as physical and environmental controls.[71]

When developing an organizational security policy, healthcare providers and EHR developers should follow a risk-based approach that encompasses the above Cybersecurity Framework functions. Risk-management strategies, tools, and techniques to inform and prioritize decisions regarding the protection of health information, including data in electronic and physical resources within its environment as well as at the point of initial collection, should be used at the front end of development of organizational security policies.

The following table lists control families that are most relevant to implementing an API in healthcare and corresponding key security policy considerations.

NIST Security Control Family	Relevant CIA Focus	Organizational Security Policy Key Considerations for API Implementation
Access Control (AC)	Confidentiality	Develop detailed baselines for implementation of practices to prevent unauthorized leakage of health information through APIs.
Audit and Accountability (AU)	Confidentiality, Integrity	Define standards of API audit logging with the appropriate fields being captured.
Contingency Planning (CP)	Availability	Verify that contingency plans include the S4S API and can ensure the API could be brought back online in the event of a disaster or interruption of service.
Identification and Authentication (IA)	Confidentiality	Provide clear guidance to developers to implement the necessary authentication configuration settings to track and verify API interactions.
System and Communications Protection (SC)	Confidentiality	Specify technical requirements that clearly define specific details for use of cryptography such as the type of key and the required cipher strength for API implementations.
System and Information Integrity (SI)	Integrity	Ensure implementers use data integrity monitoring software that could detect and help remediate unauthorized changes to health information and if left unchecked could make data unusable for the intended purpose.

[71] NIST developed the Cybersecurity Framework to help organizations in any industry to understand, communicate, and manage cybersecurity risks, including mappings to NIST 800-53 security controls. HHS OCR, in coordination with NIST and ONC, released a crosswalk that identifies "mappings" between the Cybersecurity Framework and the HIPAA Security Rule. https://www.hhs.gov/sites/default/files/nist-csf-to-hipaa-security-rule-crosswalk-02-22-2016-final.pdf

Table 2: Key Security Control Considerations for Organizational Security Policy - API Implementation

CONCLUSION & PATH FORWARD

In alignment with the 21st Century Cures Act, the use of healthcare APIs should enable individuals to better view, download, and transmit health information.[72] In the 2015 Edition Health IT Certification Criteria (2015 Edition), ONC established criteria that requires certified health IT to demonstrate the ability to provide an application access to the CCDS via an API.[73] Additionally, Stage 3 of the Medicare and Medicaid EHR Incentive Programs include specific objectives referencing the use of APIs for individual electronic access to health information and to support care coordination through individual engagement.[74]

Some in the healthcare community perceive APIs as being less secure than other components of an IT system (e.g., an EHR system), including concerns that:

- APIs may open new security vulnerabilities, with apps accessing an individual's health records with malicious intent or without receiving proper authorization; and
- APIs could provide a possible "fire hose" of data to external third parties as opposed to the "one sip at a time" access that a proprietary website or email interface may provide.[75]

Ultimately, however, as long as APIs are implemented with appropriate privacy and security safeguards in place, they can add value to individual-directed access.[76]

The key considerations gleaned from the S4SPS project help lay the privacy and security groundwork for healthcare providers and health IT developers looking to adopt APIs into their workflow. When deciding to implement an API for use in healthcare, key considerations provided in this document can serve as an informative resource for implementers to use to help them more comprehensively manage privacy and security risks in their environments. These considerations are intended to address potential privacy and security vulnerabilities and risks, and therefore, should address concerns regarding sharing health information via APIs. Specifically, API implementers should utilize protections and safeguards to ensure that:

- *Privacy*
    - HIPAA right of access is clearly enabled in the patient portal interface

[72] Section 4004 of the 21st Century Cures Act stipulates that health IT developers or entity have "published application programming interfaces and allows health information from such technology to be accessed, exchanged, and used without special effort through the use of APIs or successor technology or standards, as provided for under applicable law, including providing access to all data elements of a patient's electronic health record to the extent permissible under applicable privacy laws." See https://www.gpo.gov/fdsys/pkg/PLAW-114publ255/pdf/PLAW-114publ255.pdf.

[73] For more information, see ONC's *2015 Edition Certification Companion Guide*, *available at* https://www.healthit.gov/sites/default/files/2015Ed_CCG_CCDS.pdf and ONC's *Understanding Health IT Resource*, *available at* https://www.healthit.gov/sites/default/files/understanding-certified-health-it-2.pdf.

[74] https://www.cms.gov/Regulations-and-Guidance/Legislation/EHRIncentivePrograms/Downloads/MedicaidEPStage3_Obj5.pdf

[75] The Joint API Task Force heard testimony regarding these perceptions in January 2016 and documented these concerns in their report. https://www.healthit.gov/hitac/sites/faca/files/HITJC_APITF_Recommendations.pdf

[76] See the Joint API Task Force May 2016 recommendations, *available at*: https://www.healthit.gov/hitac/sites/faca/files/HITJC_APITF_Recommendations.pdf

    - Patient privacy preferences regarding sharing and revocation of sharing are respected
- *Security*
    - The API cannot be manipulated by invalid inputs to the API
    - The security of data in transit and health information access control mechanisms should be properly implemented in accordance with the most up-to-date industry specifications and frameworks

These considerations serve as a building block for addressing privacy and security of implementing APIs in healthcare. Additional work on FHIR-based APIs should ensure that privacy and security are increasingly built-in with functionality. As the healthcare industry continues to move toward the adoption of APIs, innovative programs, such as ONC's *Secure API Server Showdown Challenge*,[77] provide an opportunity for implementers to leverage the enclosed key privacy and security considerations and apply them toward securing novel APIs that can be used in future health information sharing activities both within an organization and with third-party apps for clinical care and medical research.

[77] https://www.cccinnovationcenter.com/challenges/secure-api-server-showdown-challenge/

GLOSSARY

Term	Definition
Application Programming Interface (API)	Technology that allows one software application to programmatically access the services provided by another software application.
Assessment	S4SPS activities that may encompass both technical testing as well as evaluation of important documentation and resources.
Common Clinical Data Set (CCDS)	A set of data elements that are required as part of the 2015 Edition certification criteria that describe an individual's demographics and associated health information about an individual.[78] EHR developers and/or healthcare providers implementing the S4S API are required to support a subset of these data elements as specified by S4S.
Credentials	Credentials are issued as the result of the registration or enrollment activity and are what users actually use, or assert they are, in order to gain access to online systems and services.[79] Examples include username and password.
Electronic Health Record (EHR)	A digital version of an individual's paper chart. EHRs are real-time, patient-centered records that make information available instantly and securely to authorized users.[80]
Endpoint	An endpoint is any component within the workflow where communication occurs with another component either internal or external to the system.
Fast Healthcare Interoperability Resources (FHIR)	An HL7-created standard for exchanging healthcare information electronically using a set of components called resources.[81]
Users and Developers of Health Information Technology (IT)	For purposes of S4SPS, this encompasses: • Healthcare providers • Health IT developers Any other entities involved in implementing or developing organizational policies and technologies.
Patient Portal	A secure website that gives individuals electronic access to personal health information (e.g., protected health information contained in a provider's EHR system) from anywhere with an internet connection.[82]

[78] https://www.healthit.gov/sites/default/files/2015Ed_CCG_CCDS.pdf
[79] https://csrc.nist.gov/csrc/media/publications/nistir/8149/draft/documents/nistir_8149_draft.pdf
[80] https://www.healthit.gov/providers-professionals/faqs/what-electronic-health-record-ehr
[81] https://www.hl7.org/fhir/DSTU2

Term	Definition
Risk	The potential for loss, damage or destruction of an asset as a result of threat exploiting vulnerabilities.
System	The combination of the following components: EHR, S4S server, patient portal, OAuth 2.0 endpoints, and the environment in which they reside.
Testing	S4SPS activities that involve direct technical evaluation methods.
Vulnerability	Flaws or weaknesses in a system that potentially expose health information.

Table 3: Glossary

[82] https://www.healthit.gov/providers-professionals/faqs/what-patient-portal