Health IT Standards Committee

A Public Advisory Body on Health Information Technology to the National Coordinator for Health IT

Transport & Security Standards Workgroup

Health IT RESTful Application Programming Interface (API) Security Considerations

March 18, 2015

Context

Background

- The TSS WG Chairs acknowledged the importance of identifying and addressing the security vulnerabilities and overall risks associated with Health IT RESTful APIs
- During the TSS WG's deliberations, the chairs of the TSS WG and the Architecture, Services, and Application Program Interface (API) Work Group (ASA WG) met and agreed upon the need to provide considerations to ONC regarding Health IT RESTful API Security
- There is no single authoritative source of security guidance or best practices for implementing the OAuth 2.0 specification
- Many open-source OAuth 2.0 libraries exist, and developers sometimes are inclined to select one without full knowledge of the security implications

Context - Continued

Pertinent Internet Engineering Task Force (IETF) Specifications:

- OAuth 2.0 Authorization Framework -- RFC 6749
- OAuth 2.0 Authorization Framework: Bearer Token Usage -- RFC 6750
- OAuth 2.0 Threat Model and Security Considerations -- RFC 6819

Considerations

- The following are some recommended topics to consider for client and browser software across multiple platforms including mobile in enabling Health IT (HIT) to be certified for implementing a secure application programming interface (API) for information sharing between partners using RESTful APIs:
  - Use OAuth 2.0 and OpenID Connect standards with transport layer security (TLS) encryption to secure HIT RESTful APIs
  - Use the OAuth 2.0 implementation model most appropriate for the architecture and risk profile of the application
  - OpenID Connect enables single sign-on across multiple applications, which increases the importance of a strong initial login – assure that the method used to initially authenticate the user is sufficiently strong for the application use case

Considerations - Continued

- The following are some recommended topics to consider for client and browser software across multiple platforms including mobile in enabling Health IT (HIT) to be certified for implementing a secure application programming interface (API) for information sharing between partners using RESTful APIs:
  - Strengthen client and browser software authentication by using standardized signed web tokens* instead of passwords transmitted over the network
  - Use TLS encryption with server side authentication to assure clients that they are communicating with the correct server and to protect data transmitted across the established link
  - Minimize the risk of data exposure through redirect manipulation by using declared redirect Unique Resource Identifiers (URIs) during client registration
  - Establish and enhance HIT RESTful API security vulnerability testing to minimize evolving cybersecurity risks

*A web token signature is a verified and secure means of representing claims to be transferred between two parties

Considerations - Continued

- The following are some recommended topics to consider for client and browser software across multiple platforms including mobile in enabling Health IT (HIT) to be certified for implementing a secure application programming interface (API) for information sharing between partners using RESTful APIs:
  - Ensure appropriate awareness and mitigation of Cross-Site API vulnerabilities
  - Vendors should provide to customers current information regarding HIT technology compatibility and interoperability with browsers and client software/platforms, and potential impacts on security
  - Vendors should incorporate threat monitoring and risk mitigation into the HIT vendor's product management lifecycle
  - ONC should also track the efforts of the OpenID Foundation Health Relationship Trust (HEART) Working Group and the Argonaut Project, both of which are addressing privacy and security for RESTful HIT APIs