Artículo de Investigación

Módulos de reautenticación para el servidor de autenticaciones Keycloak

Reauthentication modules for Keycloak authentication server

Revista sobre estudios e investigaciones del saber académico

Daicy Patricia Duarte Paiva [1]
https://orcid.org/0000-0003-0922-1072
[1]Universidad Nacional de Itapúa. Dirección de Investigación y Ambiente. Encarnación, Paraguay duarte.daicy@gmail.com

Resumen

En estos días, proteger los recursos del acceso no autorizado es primordial. Actualmente el control de acceso a recursos se realiza en base a roles (RBAC, por sus siglas en inglés) y permisos otorgados al usuario, lo cual no garantiza que sean accedidos sólo por personas autorizadas. La propuesta de este trabajo es agregar una capa más de seguridad al control de acceso en función del nivel de autenticación del usuario (en base al nivel de seguridad del autenticador definido por el NIST (Instituto Nacional de Estándares y Tecnología), es decir, un usuario puede tener un nivel de autenticación para visualizar datos, pero no así para editar datos, para realizar esta acción se requiere un nivel de autenticación mayor, por ejemplo un autenticador basado en hardware. Para lograr esto, fueron desarrollados nuevos módulos para el servidor de autenticaciones Keycloak, a fin de enviar información a la Relying Party (RP) sobre el tipo de autenticador utilizado por el usuario al momento de iniciar sesión, y a partir del dicho dato, solicitar la reautenticación del usuario en caso de que el nivel de autenticación no sea suficiente para acceder al recurso o realizar una determinada acción.

Palabras claves: Keycloak. Autenticación. Autorización. Control de acceso. Relying party.

Abstract

Nowadays, protecting resources from unauthorized access is crucial. Currently, access control to resources is carried out by roles based (RBAC) and permissions granted to the user, which does not guarantee that resources will be accessed only by authorized persons. This work proposes to add one more security layer to the access control, based on the user's authentication level (based on the authenticator security level defined by the NIST (National Institute of Standards and Technology), that is, a user can have an authentication level to visualize data, but not to edit it, to perform this action a higher level of authentication is required, for example, a hardware-based authenticator. To achieve this, new modules for the Keycloak authentication server were developed, in order to send information to the Relying Party (RP) about the authenticator type used by the user at the login time and thus request the user's reauthentication in case the authentication level is not enough to access the resource or execute a specific action.

Keywords: Keycloak. Authentication. Authorization. Access control. Relying party

Área del conocimiento: Ingeniería y Arquitectura.
Correo de Correspondencia: cemaes_py@yahoo.com
Conflictos de Interés: La autora declara no tener conflictos de intereses

Fecha de recepción: 10/12/2020 Fecha de Aprobación: 19/10/2021
Página Web: http://publicaciones.uni.edu.py/index.php/rseisa

Citación recomendada: Duarte Paiva, D. P. (2021). Módulos de reautenticación para el servidor de autenticaciones Keycloak. Revista sobre estudios e investigaciones del saber académico (Encarnación), 15(15): e2021001

INTRODUCCIÓN

Cualquier información relacionada a una persona es considerada como datos de la misma, estas informaciones pueden identificarlo como una única persona.

De acuerdo con McCallister et al., (2010), uno de los términos más utilizados para definir la información personal es el PII (por sus siglas en inglés), información que permite identificar de manera u´ nica a un individuo, estos datos incluyen desde correo electrónico, información biométrica hasta historial clínico. Hoy en día, con la digitalización, estas credenciales en línea son cada vez más valiosas ya que permiten a personas mal intencionadas obtener acceso a cuentas de terceros y de esta manera robar, desde su dinero hasta sus kilómetros de viajero frecuente, caso ocurrido en el año 2015 donde ciberdelincuentes hackearon más de 10.000 cuentas de American Airlines y United Airlines, lo cual permitió a los atacantes reservar vuelos gratuitos y obtener beneficios. Entre los robos de información personal ocurridos en los últimos años se puede mencionar el de Adobe, (2013)Yahoo Trautman & Ormerod, (2016.) y Gressin, (2017) Equifax, (2017). De acuerdo al reporte 2019 MidYear QuickView Data Breach Report según Risk based Security, (2019), en los primeros seis meses del 2019, se reportaron 3,813 infracciones exponiendo más de 4,1 mil millones de registros, y comparándolos a la cifra de mediados del 2018, la cantidad de datos expuestos aumentó un 52 %. Por lo tanto, es críticamente necesario proteger los datos del acceso desconocido y no autorizado, a través de mecanismos robustos que permitan una reautenticación con un nivel alto de seguridad a fin de evitar la suplantación de identidad, es decir, identificar con mayor seguridad a quien realmente está autorizado a acceder a los recursos o realizar alguna acción.

Pilares de la Seguridad de la Información

El estándar ISO/IEC, (2013), especifica que la seguridad de la información preserva la confidencialidad, la integridad y la disponibilidad de la información. Conocidas por sus siglas en inglés CIA (Confidencialidad, Integridad, Disponibilidad), dichos conceptos son considerados los tres pilares fundamentales de la seguridad de la información.

- **Confidencialidad**: evitar la lectura no autorizada de la información. Se debe garantizar que sólo personas auto- rizadas tengan acceso a la información. Otra definición para confidencialidad sería privacidad.
- **Integridad**: evitar que la información sea modificada por personas no autorizadas.
- **Disponibilidad**: garantizar que la información esté siempre disponible para ser accedida.

Diferencia entre Autenticación y Autorización

La autenticación es el proceso de determinar si un usuario (u otra entidad) debe tener acceso a un sistema Stamp, (2011). Es decir, es el proceso que verifica que el usuario es quien dice ser, una autenticación exitosa proporciona garantías, en base a riesgos asumidos, que el usuario que accede al servicio hoy es el mismo que accedió la vez anterior. De acuerdo con Grassi, Garcia, y Fenton, (2017) y Grassi et al, (2017) un humano puede ser autenticado en base a tres factores: algo que conoce, tiene y/o es.

Mientras que la autorización responde a la pregunta: ¿Está permitido que tú realices esta acción? La autorización está relacionada con las restricciones para realizar acciones que posee un usuario autenticado (Stamp, 2011).

Niveles de Seguridad

El Instituto Nacional de Estándares y Tecnología (NIST, por sus siglas en inglés), es el responsable del desarrollo de las directrices de seguridad de la información. Los delineamientos en la Publicación Especial NIST.SP.800-63B “Digital identity guidelines: authentication and lifecycle management” de Grassi, et al, (2017) describen tres Niveles de Seguridad del Autenticador (AAL, por sus siglas en inglés), a través de los cuales pueden ser medidos en cuanto a qué tan confiables son.

A continuación, se describen cada uno de los niveles:

AAL1: Proporciona cierta seguridad de que el solicitante controla un autenticador vinculado a la cuenta del abonado, requiere por lo menos autenticación de un factor, en inglés, Single-Factor authentication (SFA).

AAL2: Proporciona una alta seguridad de que el solicitante controla autenticador(es) vinculados a la cuenta del abonado. Pruebas de posesión y control de dos factores de autenticación son requeridos. Asimismo, a partir de este nivel, técnicas criptográficas son requeridas.

AAL3: Proporciona una muy alta seguridad, de que el solicitante controla autenticador(es) vinculados a la cuenta del abonado, pruebas de posesión a través de protocolos criptográficos y dos factores de autenticación distintos son requeridos. En este nivel se requiere además de una autenticación basada en hardware que proporcione resistencia a la suplantación de identidad.

Keycloak

Keycloak es un servidor de autenticaciones lanzado en el 2014, está desarrollado en JAVA y es open-source bajo la licencia Apache2 (Keycloak, 2019a).

Permite que desarrolladores de aplicaciones no tengan que preocuparse por el ingreso (log in), autorización o la página de registro de usuarios. Keycloak ofrece dicho servicio, además a través del inicio de sesión único (SSO, por sus siglas en inglés) permite a los usuarios tener acceso a diferentes sistemas autenticándose so´ lo una vez, a trave´s de protocolos OpenID Connect y SAML. Keycloak proporciona una serie de Service Provider In Terface (SPI) para los cuales es posible implementar nuestros propios proveedores (providers, en inglés) es decir, desarrollar plugins que se pueden instalar en el servidor.

METODOLOGÍA

Flujo de Autenticación

Para el desarrollo de un nuevo proveedor es necesario entender los siguientes términos, según la documentación oficial de Keycloak (Keycloak, 2019b). Como se muestra en la figura 1, en la pestaña Flujos (Flows, en inglés), se pueden observar todos los Auth Type definidos y cada una de sus configuraciones, donde Auth Type es el nombre de la autenticación o acción que se ejecutará de acuerdo a la prioridad de ejecución (Executions Requirements, en inglés).

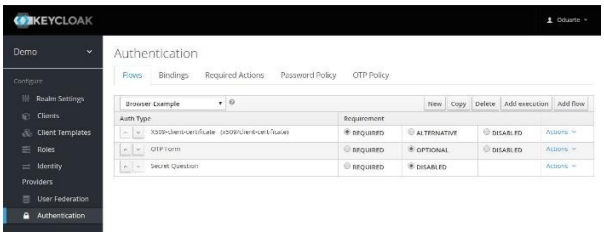
Flujo de autenticación (Authentication Flow, en inglés), es un contenedor con todas las autenticaciones que deben ocurrir, como se puede observar en la figura 1, en este caso Browser-Example es el nombre del contene- dor. Para cada Auth Type, un requisito de ejecución (Executions Requirements) debe ser configurado. Los Executions Requirements definidos en Keycloak son los siguientes:

- REQUERIDO, se debe ejecutar siempre exitosamente.
- ALTERNATIVO, al menos uno de los autenticadores configurado como alternativo debe ejecutarse exitosamente. Una vez que uno de los que están configurados como alternativo se ejecuta, los demás ya no se ejecutan.
- OPCIONAL, se ejecuta solo si el usuario tiene configurado ese tipo de autenticador, caso contrario se ignora.
- DESABILITADO, no se ejecuta.

- *Authenticador*, es un componente que contiene la lógica para realizar la autenticación o la acción requerida.
- *Ejecución*, es un objeto que une el autenticador al flujo (a través de los requisitos de ejecución).
- *Requisitos de ejecución*, define como se comporta un autenticador dentro del flujo de autenticación.
- *Acción requerida*, una acción que el usuario debe realizar después de la autenticación, pero una vez que la acción es exitosa. Restablecer una contraseña o la verificación del correo electrónico son ejemplos de una acción requerida.

Figura 1.

Ejemplo del Flujo de Autenticación



RESULTADOS

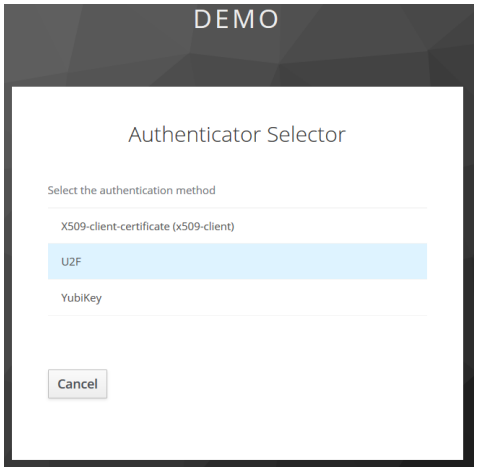
Módulo de Ingreso (Log In)

Para poder seleccionar el tipo de autenticador que se utilizará en el momento del inicio de sesión, es necesario personalizar el flujo de autenticación y crear una implementación personalizada del autenticador SPI para poder enviar esta información dentro del token, ver sección VIII módulo mapeador de protocolo (protocol mapper, en inglés).

En este módulo, se genera una lista con opciones de métodos de autenticación habilitados y que puede utilizar el usuario para ingresar, ver figura 2. Una vez seleccionada la opción de autenticación, la manera en que se almacena esa información es a través de la interfaz proveída por Keycloak UserSessionModel, la cual permite obtener datos del usuario y almacenar información del método de autenticación que utilizó para ingresar como un atributo de la clase, de esta manera se disponibiliza para ser accedida desde el módulo protocol-mapper, que se encarga de enviar información del tipo de autenticador a la Relying Party (RP).

Figura 2.

Nuevo módulo que permite seleccionar el método de autenticación



La lista de opciones se obtiene a partir de los requerimientos de ejecución. Se agregan a la lista todos los tipos de autenticadores que no se encuentren deshabilitados.

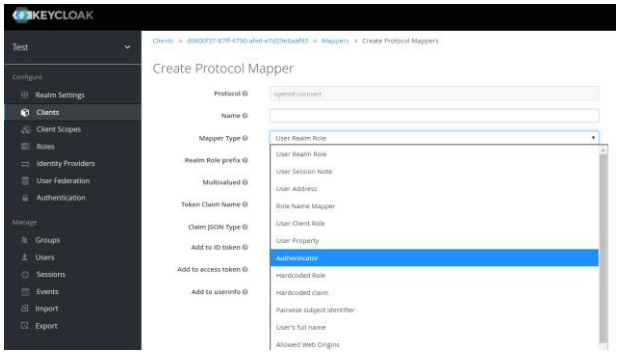
Módulo Mapeador de Protocolo (Protocol Mapper)

En ocasiones es posible que las aplicaciones deseen o necesiten enviar ciertos metadatos dentro del token OpenID Connect (OIDC) o la aserción SAML, esto es posible mediante el módulo protocol- mapper. Keycloak ya cuenta con informaciones predeterminadas que se envían dentro del token, estas informaciones son la dirección de correo electrónico y el nombre del usuario, como se puede observar en la figura 3. La configuración del protocol-mapper se realiza por cliente. En este caso es necesario enviar el tipo de autenticador que utiliza el usuario al ingresar. Esta información es importante para la Relying Party (RP) porque de acuerdo a la información del tipo de autenticador, y dependiendo del nivel de autenticación, el usuario puede o no acceder a ciertos recursos. En caso de que el método de autenticación utilizado no cuente con un nivel de seguridad aceptable para dicho recurso, el usuario debe volver a autenticarse con el tipo de autenticador que el recurso solicita para tener acceso a la información.

En Keycloak, dentro del menú cliente se tiene la opción de configurar otros mapeadores que ya están definidos como el de roles y atributos personalizados. En este momento, la opción de enviar información del tipo de autenticador utilizado por el usuario para iniciar sesión no está disponible, por esta razón es necesario desarrollar un nuevo módulo protocol-mapper personalizado y de esta manera enviar dicha información dentro del token.

Figura 3.

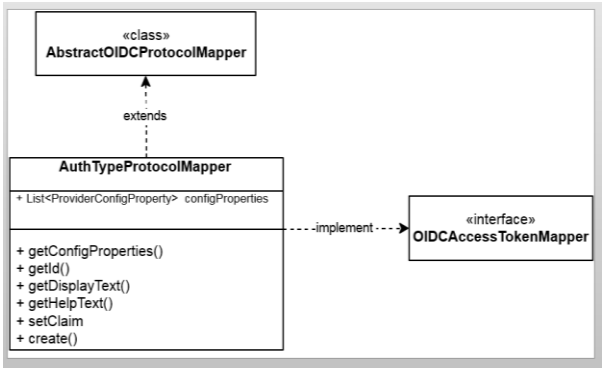
Configuración del protocol mapper.



Para obtener y enviar la información del tipo de autenticación utilizado por el usuario, primeramente, es necesario instalar el nuevo plugin, para que el nuevo tipo de mapeador (Authenticator) se encuentre disponible en la lista de opciones de Keycloak, como se muestra en la figura 3. Para esto, es necesario crear una nueva clase que extiende de AbstractOIDCProtocolMapper y a través del método ProviderConfigurationBuilder.create() responsable de construir la lista de configuraciones de metadatos de propiedad (ProviderConfigProperty), el cual se utiliza para representar páginas de configuraciones genéricas para extensiones de Keycloak en la consola de administración. Además, implementar la clase OIDCAccessTokenMapper, que permite agregar atributos al token de acceso. En la figura 4 se puede observar una distribución de la nueva clase AuthTypeProtocolMapper, esta clase se encarga de enviar información del tipo de autenticador utilizado por el usuario al momento de ingresar.

Figura 4.

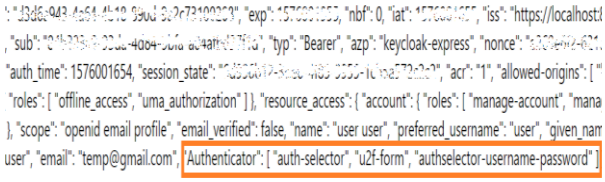
Descripción general de la clase AuthTypeProtocolMapper



Keycloak ofrece el método setClaim, que permite asociar el valor de un atributo al token, de esta manera se asocia el tipo de autenticador utilizado por el usuario para loguearse y se envía dicha información a la relying party (RP) dentro del token, como se observa en la figura 5.

Figura 5.

Información del token recibido por la relying party (RP)]



Desarrollo de la Relying Party

Keycloak provee adaptadores o librerías para asegurar clientes y servicios, dependiendo de la plataforma. En este caso, es necesario el adaptador JavaScript ya que para el desarrollo de la Relying Party (RP) se utilizó ReactJS para las interfaces y NodeJS para el desarrollo del servidor.

De acuerdo a la documentación de Keycloak, (2019b) sus adaptadores de cliente son librerías que hacen que sea más sencillo asegurar aplicaciones y servicios con Keycloak. Keycloak provee adaptadores para diferentes plataformas, entre ellas JavaScript (client-side) y NodeJS (server-side).

Este cliente especifica exactamente qué tipo de autenticador es necesario para acceder a cada recurso, de acuerdo a los niveles de autenticación del usuario. En caso que el usuario se autentique con un autenticador cuyo nivel de seguridad sea menor al necesario para acceder al recurso, se obliga al usuario a que vuelva a autenticarse especificando el tipo de autenticador requerido. Para el caso de OIDC (por sus siglas en inglés), esto se logra a través del parámetro prompt=login.

CONCLUSIÓN

Hoy en día, una fuerte cultura de protección de la información es necesaria en las organizaciones. Aunque es sabido que las contraseñas no son seguras, siguen siendo el método más utilizado para proteger el acceso a los recursos de un sistema (Bonneau et al., 2012). Además, existen varias herramientas cada vez más sofisticadas para descifrar contraseñas, que tardan minutos u horas dependiendo del hardware (De Carné de Carnavalet & Mannan, 2014)

Sin embargo, es necesario recalcar que no todas las cuentas o recursos necesitan un autenticador fuerte, por ejemplo una cuenta desechable para realizar una compra sin guardar las credenciales de pago o una cuenta de suscripción a un periódico, en relación a cuentas que contienen datos de valor, cuya exposición pública o eliminación pueden tener consecuencias graves (Grosse & Upadhyay, 2013). Por esta razón, contar un mecanismo de protección basado en una doble autenticación, y basado en hardware es altamente recomendado.

Finalmente, el resultado de esta investigación sugiere que debe agregarse una capa más de seguridad al control de acceso a los recursos en función del nivel de autenticación del usuario, utilizando autenticadores de un nivel de seguridad aceptable para poder proteger los recursos que contienen principalmente información sensible complementando el uso de contraseñas para este propósito, ya que al utilizarlas como un único factor de autenticación no brindan una seguridad completa de que el solicitante controla el autenticador vinculado a su cuenta.

REFERENCIA BIBLIOGRÁFICA

Adobe Blog. (2013, Marzo). *A. Blog.* Retrieved from Important customer security announcement : https://theblog.adobe.com/important-customer- security-announcement

Bonneau, J., Herley, C., Van Oorschot, P. C., & Stajano, F. (2012). “The quest to replace passwords: A framework for comparative evaluation of web authentication schemes,” in Security and Privacy (SP). *Symposium on. IEEE, 2012*, 553–567.

De Carné de Carnavalet, X., & Mannan, M. (2014). From Very Weak to Very Strong: Analyzing Password-Strength Meters. *Network And Distributed System Security Symposium*, 23-26.

Equifax. (2017). *Consumer Notice*. Retrieved from 2017 Cybersecurity Incident & Important Consumer Information: https://www.equifaxsecurity2017.com/consumer-notice/

Grassi, P. A., Fenton Elaine, J., Newton, L. M., Perlner, R. A., Regenscheid, A. R., Burr, W. E., . . . Theofanos, M. F. (2017). *Digital identity guidelines: authentication and lifecycle management.* Gaithersburg:

National Institute of Standards and Technology Special Publication 800-63B. Retrieved from https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63b.pdf

Grassi, P. A., Garcia, M. E., & Fenton, J. L. (2017). *Digital identity guidelines: revision 3*. Gaithersburg. Retrieved from https://nvlpubs.nist. gov/nistpubs/SpecialPublications/NIST.SP .800-63- 3.pdf

Gressin, S. (2017, setiembre 8). *Federal Trade Comission. Consumer Information*. Retrieved from The Equifax Data Breach: What to Do, Federal Trade Commission, Washington, DC : www.consumer.ftc.gov/blog/2017/09/equifax-data-breach-what-do

Grosse, E., & Upadhyay, M. (2013). Authentication at scale. *IEEE Security and Privacy*, *11*(1), 15–22.

ISO/IEC, ". 2. (2013). *Information technology — Security techniques — Information security management systems.*

Keycloak. (2019a). *Keycloak*. Retrieved from Open Source Identity and Access Management For Modern Applications and Services: https://www.keycloak.org/

Keycloak. (2019b, Junio 15). *Keycloak*. Retrieved from documentation: https://www.keycloak.org/documentation.html

McCallister, E., Grance, T., & Ken, K. (2010). "Guide to protecting the confidentiality of personally identifiable information (PII),. *Special Publication 800-122 Guide*, 1–59.

Risk based Security. (2019). 2019 midyear quickview data breach report. *Cyber Risk Analytics*, 1-14.

Stamp, M. (2011). *Information security: principles and practice.* (Second ed.). Wiley.

Trautman, L. J., & Ormerod, P. C. (2016.). Corporate Directors' and Officers' Cybersecurity Standard of Care: The Yahoo Data Breach. *American University Law Review, 66*(5), 1232-1291. Retrieved from https://digitalcommons.wcl.american.edu/aulr/vol66/iss5/3