

EXTENSIBLE KEY MANAGEMENT SYSTEM FOR APPLICATION PROGRAM INTERFACES

PRIORITY

[0001] This application claims the benefit of U.S. Provisional Application No. 62/473,194 filed March 17, 2017, which is hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] This disclosure relates to application program interfaces, and more particularly, to an extensible key management system for application program interfaces.

BACKGROUND

[0003] Digital products and services are increasingly interconnected, delivering seamless experiences to end-users. One method of delivering this interconnectivity globally is with Application Program Interfaces (APIs), which are typically a set of routines, protocols and tools that determine how two software components interact with each other. Developers spend significant amounts of time sourcing and building relationship with API providers and will do so even more as the market for API usage grows ever larger. There is a need for systems and methods to reduce this administrative overhead.

SUMMARY

[0004] An extensible key management system for application program interfaces (APIs) provides a platform to consolidate, aggregate and simplify API keys, from API providers, such as from financial institutions and other digital delivery services. The system provides a platform with a collection of APIs in a single place which may dramatically

reduce the development overhead associated with building API driven propositions and managing associated API keys.

[0004a] According to a first aspect of the present invention, there is provided a method of performing extensible key management for application program interfaces (APIs), the method comprising: accessing, by a processor, an application via a developer services portal; identifying, by the processor, initiation via the developer services portal of an association of the application with an external API of a third party provider service; obtaining, by the processor, a sub-key for the external API from the third party provider service, authorization to obtain the sub-key based on a master key previously provided from the third party provider service; mapping, by the processor, an association between the sub-key, the application and the external API of the third party provider service; and replacing, by the processor, a developer API key included in a service call message with the sub-key, the service call message resulting from a request from a user device executing the application.

[0004b] According to a second aspect of the present invention, there is provided an extensible key management system comprising: an API proxy associated with an application and executable by a processor to proxy a service call message for an external API of a third party provider service; a context service executable by the processor to parse the service call message and identify a developer API key included in the service call message; a key management service executable by the processor, in response to identification of the developer API key, to replace the developer API key with a sub-key for the external API, the sub-key being associated with the developer API key, the application and the external API of the third party provider service; and the key management service further executable with the processor to proxy the service call message with the sub-key to the external API of the third party provider service.

[0004c] According to a third aspect of the present invention, there is provided a method of performing extensible key management for application program interfaces (APIs), the method comprising: associating, via a developer services portal, an application with a first API proxy for a first external API of a first third party provider

service and with a second API proxy for a second external API of a second third party provider service; obtaining and storing, by a processor, a first sub-key for the first external API and a second sub-key for the second external API when the application is associated with the first API proxy and the second API proxy; receiving a first service call message from the application for the first external API that includes a developer API key and receiving a second service call message from the application for the second external API that includes said developer API key; replacing, by the processor, the developer API key with the first sub-key in the first service call message based on a stored association of the developer API key with the first sub-key, the application and the first external API; and replacing, by the processor, the developer API key with the second sub-key in the second service call message based on a stored association of the developer API key with the second sub-key, the application and the second external API.

[0005] The system may enable an intelligent, configuration based approach to managing the centralized consumption of API products from multiple sources.

Historically it has been easy to create a solution that aggregates or selects over a small set of similar APIs by building services or components that physically implement logic to do the aggregation (calling many services but producing a single set of results) or selection (choosing a specific service out a large group of similar services based on a set of criteria). However, this becomes increasing unwieldy to manage, maintain and grow as the number of services increases or the services vary in form (e.g. number/style of inputs or output formats). It is also very difficult, time consuming and expensive to deal with downstream changes to APIs being aggregated over as well as to add new services to the pool and often requires code releases or similar release management processes to maintain a stable and working system.

[0006] The extensible key management system can solve this problem by allowing a layered, stateless, configuration-based approach to the management of API services and API keys, which may enable aggregation, selection and proxy services with

centralized key management capabilities that can be quickly and efficiently modified, extended and maintained via run-time modifiable stored configuration entities.

[0007] Other systems, methods, features and advantages will be, or will become, apparent to one with skill in the art upon examination of the following figures and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and the following claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The embodiments may be better understood with reference to the following drawings and description. The components in the figures are not necessarily to scale.