

API Integration Tutorial: Testing, security and API management



In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

In this tutorial:

As application program interface integration increases, so do the challenges with maintaining management, testing, and security.

This API integration tutorial compares leading API management tools currently available on the market, as well as strategies for RESTful API testing.

Get a better understanding of API integration and management strategies.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

■ A roundup of the leading API management tools available today

Zachary Flower, Freelance writer, SearchMicroservices.com

API management is a constantly growing market with more new products popping up every year. As a result, narrowing all of the potential choices down to one perfect platform can feel like an overwhelming task. The sheer number of options out there makes choosing one an extremely tough decision.

To help relieve the strain involved with this decision, we've put together this detailed product roundup covering 10 of the leading API management tools currently available on the market. These overviews, appearing in alphabetical order, will provide you with key information you need when deciding which of these platforms is right for your organization. They'll also provide you with a solid understanding of the state of today's [API management](#) marketplace.

3scale API Management

- Cost:** Starting at \$50 per month
- License:** Commercial
- Market:** Startup/SMB/Enterprise
- Delivery:** Proxy/Agent/Hybrid

In this tutorial

- [A roundup of the leading API management tools available today.....](#) p.2

- [What are some solid options for open source API management tools?.....](#) p.14

- [The basics of establishing a RESTful API testing program](#) p.17

- [Testing microservices and APIs in the cloud.....](#) p.24

- [How do I create a secure API for mobile?.....](#) p.29

- [About SearchMicroservices.com](#) p.31

3scale is an incredibly flexible API management platform from [Red Hat Inc.](#) Unlike many of the API management tools on this list, 3scale has a very self-serving product offering, which has resulted in a very affordable and flexible platform. The low barrier-to-entry of the 3scale platform allows organizations to slowly scale up their API(s) both architecturally *and* financially, rather than having to make a high-cost commitment on a brand new service. Although 3scale does offer both on-site and cloud-based delivery methods for the API management platform, its particular brand of delivery is unique in that the on-premises gateway is built around the NGINX server. This gives organizations the flexibility to use whatever infrastructure best suits their needs. Additionally, 3scale integrates directly with popular cloud providers Heroku, [Amazon Web Services](#) and Red Hat OpenShift for more rapid deployments.

While 3scale has incredibly flexible delivery methods, the feature that stands out the most is the end-user experience. 3scale operates under the philosophy that API success is dependent on creating an awesome developer experience. This attitude has led to the creation of engaging developer portals and live [Swagger-based](#) documentation for a dynamic, usable developer experience.

Starting at \$50 per month, 3scale's base package is incredibly financially approachable. This base package includes the most desirable features in an API management platform, including rate limiting, analytics, a developer portal and key management. The next package up from there, starting at \$150 per month, adds API monetization and multiple apps per account to the list of features. The final tier, priced at \$750 per month, is aimed at larger organizations with multiple APIs, users and administrators. 3scale does offer

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

enterprise-level features and pricing as well, but those interested should contact the company for pricing information.

Akana Platform

- Cost:** Starting at \$4,000 per month
- License:** Commercial
- Market:** Enterprise
- Delivery:** Proxy/Agent/Hybrid

Akana Platform from Rogue Wave Software is an enterprise-level API management system aimed at larger organizations with highly defined product lifecycle workflows. Touting customers like American Express, Verizon and eBay, Akana Platform comes equipped with a developer portal and tools for design, security, traffic management, mediation, integration, analytics and monetization. Akana stands out, however, in its collection of [lifecycle management tools](#). It provides a version control system that allows for approval-based changes, and integrates directly with popular DevOps platforms like Jenkins, JIRA and GitHub to keep everyone in the loop.

It's also worth noting that Akana Platform is a gold-certified Microsoft partner, meaning the company offers a genuine, Microsoft-approved system for enterprises developing products on the Microsoft technology stack. All of Akana Platform's tools integrate directly with Microsoft's cloud platforms, which includes built-in support for [Microsoft Visual Studio](#) and Team Foundation Server within its Lifecycle Management toolset. There's even native Windows Server and Azure cloud support. In addition to version

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

control and DevOps integrations, the platform also provides a low-level relationship visualizer, allowing stakeholders to visualize how all the pieces of an API relate to each other, providing full visibility into how every change affects the architecture of the entire system.

Apigee

- Cost:** Starting at \$300 per month
- License:** Commercial
- Market:** SMB/Enterprise
- Delivery:** Proxy/Agent/Hybrid

Apigee is a transparency-driven API management tool from Apigee Corp. that is aimed at small to medium-sized businesses, but with substantial enterprise-level options as well. The Apigee platform has some strong analytics tools, allowing organizations to track API performance, latency, error rates, traffic and usage, and even provides a real-time debugger for tracing issues down to the source.

While Apigee has some great analytics tools, what really sets it apart are its monetization tools. With Apigee, organizations can monetize their APIs in a multitude of unique ways, from offering free trials based on time or API call volume to charging users based on usage or a flat fee over time. However, Apigee has added a truly unique twist to this feature: revenue-sharing capabilities that allow organizations to share revenue with their own API consumers. This gives organizations the ability to not only charge a percentage of revenue generated by API consumers, but also gives that

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

same organization the ability to share a percentage of revenue with API consumers. This can potentially open the door for powerful affiliate programs and partnerships.

Azure

- Cost:** Starting at \$50 per month
- License:** Commercial
- Market:** Startup/SMB/Enterprise
- Delivery:** Proxy

Microsoft is a relative newcomer to the field of API management, especially when compared with a few of the other API management tools in this roundup. The [Azure API management platform](#) is a cloud-based gateway targeted toward organizations of all sizes, starting at just \$50 per month and scaling up as needed. While the Azure API management platform will work with any host via a direct or virtual private network connection, the big advantage to using the platform is its direct integration with the rest of the Azure ecosystem.

With direct support for services like the Azure Active Directory and the ability to work seamlessly with Azure virtual networks and ExpressRoute, the Azure API management platform is the perfect choice for organizations that are already consuming the Azure ecosystem. Beyond the obvious integrations within its own architecture, getting started with the Azure API management platform is as easy as importing an existing Swagger

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

specification. This gives organizations the flexibility to move to Azure from any other platform with very little overhead.

CA API Management

Cost: Not provided
License: Commercial
Market: Enterprise
Delivery: Cloud/Agent/Hybrid

CA Technologies' API management platform is geared heavily toward enterprise organizations, resulting in a very sales- and consulting-driven organization. As a result of this enterprise-leaning sales funnel, pricing for the CA API Management platform is not readily available.

While the CA API Management platform shares many of the same features as other offerings, such as a developer portal, endpoint management and analytics, the thing that makes it stand apart is its emphasis on mobile API management. The feature that highlights this support is the CA Mobile API Gateway. This is an API gateway that provides [mobile-friendly API design](#) and is specifically tailored to handle things like security, identity, adaption, optimization and integration. By providing software development kits (SDKs) and granular security policies, the CA Mobile API Gateway can drastically speed up application development and deployment, while also ensuring a [safe and secure](#) API environment.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

It is important to note that these features focus specifically on mobile-first use cases through increased caching, mobile-specific SDKs and standards-based architecture and authentication methods, making the CA Mobile API Gateway stand out from standard API gateways.

Cloud Elements

- Cost:** Starting at \$1,000 per month
- License:** Commercial
- Market:** Startup/SMB/Enterprise
- Delivery:** Proxy

Cloud Elements Inc. offers users a different sort of API management platform in that its APIs are built around the concept of connecting disparate data sources and services through entities called hubs and elements. What makes this concept unique is that its namesake Cloud Elements platform integrates with a multitude of third-party APIs and services (the aforementioned elements), allowing organizations to build out complex REST APIs with minimal engineering overhead. A great analogy of this hub-and-element model (albeit a simplistic one) is that Cloud Elements is like the IFTTT of APIs.

Cloud Elements executes this integration strategy by providing data transformation tools and workflow orchestration, providing total control over how your organization's internal and external services interact with each other within your API. By providing a centralized management system, Cloud Elements enables organizations to track the usage, response time and even

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

logs across all of these platforms. This allows development teams to spend less time dealing with the nuances of third-party API integrations and more time building and managing a reliable product. With very little effort, an existing API can be integrated directly with Salesforce, SendGrid, Slack and even [JIRA](#), making it possible for organizations to use the tools they are already familiar with to manage the health of their API.

Kong

- Cost:** Starting at \$0 per month
- License:** Apache 2.0
- Market:** Startup/SMB/Enterprise
- Delivery:** Proxy

Mashape Inc., like Cloud Elements, handles API management much differently than the other leading platforms. Its core offering, Kong, is an open source API gateway that can run in front of any RESTful API. This is similar to the cloud-based delivery methods of most commercial API management tools -- the main difference being that the proxy is managed internally rather than externally. Kong has a highly extensible [plug-in architecture](#), which enables total control over the features and services provided by your API. At a high level, the officially supported plug-ins handle things like authentication, security, traffic control, analytics and even data transformations and logging. This plug-in-oriented feature set makes it possible to support only the features that are important to your project, without having to worry about additional features and services going unused.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

Though not open sourced like Kong, Mashape also offers a powerful commercial API analytics platform called Galileo and a commercial developer portal platform called Gelato. These two services can move an existing API from the private to the public space by adding tools that are important to product stakeholders and API consumers. Galileo accomplishes this with analytics tools that provide real-time logging of API calls, request replaying, usage statistics and even error alerting. These are desirable features for product managers and development teams alike, as they give total insight into what is and isn't working within an API. Gelato, on the other hand, provides a drop-in public face to an API. With Swagger and [API Blueprint](#) support, developer registration management, a live API explorer, and even markdown management of [API documentation](#), Gelato is an excellent developer portal product.

Mashery

- Cost:** Not provided
- License:** Commercial
- Market:** SMB/Enterprise
- Delivery:** Proxy/Agent/Hybrid

Mashery from TIBCO Software Inc. is an API management platform designed and built with ease-of-use in mind. While not as enterprise-leaning as other platforms, Mashery is still a good fit for small to medium-sized businesses that are beyond the early startup stage. This is evidenced through TIBCO Software's direct [sales-driven funnel approach](#), which is a change from the more organic, self-serve onboarding process found

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

amongst other platform providers. Despite the larger business-size market, TIBCO has built out an impressive support system for its customers, including everything from white papers to webinars and events, as well as strategy and technical support services.

What makes Mashery so unique is its API Microflow designer, which is, by far, the platform's standout feature. At a high level, the Microflow designer is a browser-based tool that can be used to describe the operational behavior of an API. Individual Microflows can be used to manage data transformations, request routing, query throttling and even implement caching. These policies can be inserted directly in the path of the request or response workflow, allowing users to transform individual endpoint responses, throttle requests for abused operations and cache infrequently updated responses.

MuleSoft

- Cost:** Not Provided
- License:** Commercial
- Market:** Enterprise
- Delivery:** Proxy/Agent/Hybrid

MuleSoft is an enterprise-level service that goes beyond providing just an API management tool. The crux of MuleSoft's service is its [Anypoint Platform](#), a service that acts as the glue between disparate services and APIs. While MuleSoft Inc.'s API management tool is part of the Anypoint Platform, the platform itself is a collection of tools and services that allow

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

organizations to compose their API architecture and integrate it with a multitude of third-party APIs and services. It also provides a marketplace for easily finding and integrating custom elements (such as connectors, templates and APIs).

MuleSoft is an excellent platform for organizations that need to quickly and easily integrate with multiple third-party services. But what sets MuleSoft apart is the large number of *other* add-ons available in the Anypoint Exchange marketplace. This marketplace not only provides third-party API integrations, but also connectors, templates and examples. This means users can easily add Salesforce and Slack through the third-party API integrations, as well as implement real-time synchronization of users and Salesforce customers through the "Salesforce to Database User Broadcast" template.

WSO2

- Cost:** Starting at \$0 per month
- License:** Apache 2.0
- Market:** Startup/SMB/Enterprise
- Delivery:** Proxy/Agent/Hybrid

WSO2 is, by far, the most popular and feature-complete open source API management platform on the market today. It provides everything from design and prototyping tools to developer tools and traffic management. While WSO2 is loaded with the same features found in the other leading API management tools, what really makes it special is the

In this tutorial

- [A roundup of the leading API management tools available today..... p.2](#)
- [What are some solid options for open source API management tools?..... p.14](#)
- [The basics of establishing a RESTful API testing program p.17](#)
- [Testing microservices and APIs in the cloud..... p.24](#)
- [How do I create a secure API for mobile?..... p.29](#)
- [About SearchMicroservices.com p.31](#)

customizability of the platform. [WSO2 is highly configurable](#), allowing organizations to modify the look, feel and usage of the platform through styling, theming and code extensions. These extensions can be developed internally or found as additional open source projects that are built and managed by other WSO2 consumers.

Because WSO2 is open sourced, it's worth bringing attention to the increased security and quality that often comes with open source software. Popular open source projects often have hundreds to thousands of developer eyes on them, allowing for issues to be caught and fixed much earlier than many other competitors. This security benefit is known as "Linus' Law," named for the creator of Linux, Linus Torvalds. [Linus' Law](#) states that "given enough eyeballs, all bugs are shallow."

Wrapping up

While the products listed above are well-known and well-supported, they are just a few of many. As such, it's crucial to do your own research and collaborate with your team to find the software that works best for the needs of you and your team. If one of the products above does not quite fit the bill, then use the closest one as a starting point in your search for the perfect API management platform.

In the end, the most important thing to remember when evaluating any new DevOps platform is that getting the buy-in from your entire team determines success far more than how many bells and whistles any one platform has.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

■ What are some solid options for open source API management tools?

Twain Taylor, Freelancer, SearchMicroservices.com

APIs are the fuel that powers all interactions between applications behind the scenes. APIs have become essential to enabling services to talk to each other, especially as more enterprises shift to distributed architectures, like microservices.

While many tend to go the commercially supported API route and make use of [proprietary tools](#), an increasing number of organizations are making use of publicly available open source tools. Let's look at the top open source API management tools available today

Kong

Created by Mashape, Kong acts as a management layer for all your APIs. It sits between the microservices that contain the API agents and the load balancer that routes requests across all APIs. Powered by Nginx, Kong excels at distributing API tasks, whether on premises or in the cloud. It can be set up in a single or multi-data center environment and is compatible with any modern infrastructure stack.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

Based on [plug-ins](#), Kong is easy to extend and has ready-made plug-ins for many popular services. For example, it can be used to manage and invoke Amazon Web Services Lambda functions, and monitoring data can be sent to a monitoring tool, like Datadog, or a logging service, like Loggly.

Tyk

Tyk is another capable open source API management tool. It can be set up by yourself on premises, or you can pay to use a version hosted by Tyk. A recently launched Kubernetes integration also lets you set up Tyk in a Google Cloud Kubernetes container cluster. Central to the Tyk experience is its API gateway; it's actually Tyk's API that lets you manage your own API requests.

Tyk is evolving quickly, and you'll need to keep a close watch on all its releases since many come with major updates. Tyk claims that v2.3 can handle more than double the traffic of v2.2. This fast pace of development makes it hard to document every change adequately. Some very new features may not be included in the main product documentation and are found in the release notes. While Tyk does report on performance and error metrics, it could do better at showing the details of the errors, which can help with [root cause analysis](#).

Like Kong, Tyk is also easy to extend. There are already integrations with many tools, such as logging integration with Logstash. You can also write plug-ins for Tyk in a range of languages, including gRPC and Python. You

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

should plan on eventually integrating plug-ins into the open source software core of Tyk.

API Umbrella

Developed by the National Renewable Energy Laboratory, API Umbrella powers the popular data.gov website. API Umbrella provides all the standard API management features, including key management, rate limiting and analytics. It features a web admin interface and has well-indexed documentation.

Unlike the previous open source API management tools, which feature commercial hosted options, API Umbrella only has an open source option. This means you will be fully dependent on the community for support. However, there is an active community backing this project, and new releases are quite frequent.

A buzzing market

Clearly, the market for open source API management tools is buzzing with activity, and you'd be spoiled for choice trying to pick the best among them. In addition to the tools mentioned above, you should also consider taking a look at WSO2, ApiAxle and Fusio offerings.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

■ The basics of establishing a RESTful API testing program

Greg Sypolt, Senior Engineer, Gannett

The standard practice of testing web applications for many years was heavily front-end graphic user interface tests. Recently, the standards of testing have undergone a makeover from top to bottom. To move toward a modern delivery pipeline, such as continuous integration or continuous delivery, teams need to be at least versed in automation, and focus must shift upon preventing bugs by building in quality upfront and efficiently. Developers, product managers, business analyst and QA team members must begin by planning and building a better testing portfolio. As they start working closely together, they will begin to identify what they need to meet the acceptance criteria and determine as a team what the best way to test is. This also requires following the "testing pyramid methodology" by creating more lower-level tests -- API tests, for instance -- creating a smaller percentage of UI tests in the end.

This journey starts with [RESTful API](#) testing. At the end of this article, you will have a better understanding of why API testing is necessary, its key physical and logical layers, how to design a testing blueprint for APIs, and the technologies available to help.

In this tutorial

- [A roundup of the leading API management tools available today.....](#) p.2

- [What are some solid options for open source API management tools?.....](#) p.14

- [The basics of establishing a RESTful API testing program](#) p.17

- [Testing microservices and APIs in the cloud.....](#) p.24

- [How do I create a secure API for mobile?.....](#) p.29

- [About SearchMicroservices.com](#) p.31

What is a RESTful API?

In simple layman's terms, **REST** is a way for two computers to communicate with each other over the internet. One acts as a web browser, and the other acts as a web server. The transfer happens over the web, which communicates through HTTP/HTTPS protocols. The HTTP request methods performed are GET, PUT, POST and DELETE.

The two commonly used methods are GET and POST:

- **GET** is used for retrieving data from a specified resource.
- **POST** is used for creating data to be processed to a specified resource.

An example of this is the GitHub API for repository_url, shown below: The moment the HTTP request is fired it will begin communicating with the layered architecture.

Review the REST Layered Architecture

The REST layered architecture is simply a **software design pattern** that is implemented in a stack of layers. The traditional layers help you better organize the code. For instance, an application can have the following layers: presentation, business Logic, data access and data storage.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

Presentation

The top-most layer of the application is the user interface, which translates the response into machine readable format for users to understand. The presentation layer takes care of content representation, shown here:

Business Logic

The logical layer is the brains of the application. This layer processes and communicates the data between the layers and makes logical decisions. Some "logical" decisions the logical layer makes include the handling of the API, authentication and authorization, auditing, and exception handling.

Data Access

The data access layer provides simplified access to data stored -- database, flat files and more -- separate from business logic and presentation code. It is tightly coupled with the data storage layer.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

Data Storage

The data storage layer will consist of a relational database management system, NoSQL database, file system, remote storage in the cloud or in-memory.

Design an API Blueprint for Development and Testing

When designing your RESTful API testing strategy blueprint, it is important to outline all the activities that will be carried out for a sprint and the release. Having a clear testing strategy will increase the collaboration and communication early in the sprint, rather than towards the end. Often, teams see an uptick in productivity when everyone on the team takes on more ownership, responsibility and accountability for product quality.

User experience pyramid

When launching and maintaining an API, the entirety of the team needs to consider every aspect of the API user experience. This pyramid represents the different API characteristics the team should consider:

In this tutorial

- [A roundup of the leading API management tools available today.....](#) p.2

- [What are some solid options for open source API management tools?.....](#) p.14

- [The basics of establishing a RESTful API testing program](#) p.17

- [Testing microservices and APIs in the cloud.....](#) p.24

- [How do I create a secure API for mobile?.....](#) p.29

- [About SearchMicroservices.com](#) p.31

Selecting a Testing Technology

The options can be overwhelming; there are lots of open source tools, vendor tools and custom testing harnesses. It's useful to think practically about your requirements while evaluating any testing technology.

Test Coverage

To build quality products that everyone raves about requires a well thought-out testing strategy. It starts early in the process of writing code by writing tests in parallel during the sprint. This will lead to better design and fewer bugs. Unit tests are the backbone of any test strategy.

Defining Standards

It's critical to establish standards upfront and communicate them to everyone on the team. This will help avoid, for instance, general testing and coding headaches caused by unexpected changes or inability to scale. Continue to evaluate and recalibrate along the way.

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

Responsibilities

Clearly outline each team member's responsibilities for developers, QA, DevOps teams and product managers.

Continuous Testing

Continuous testing is not only about the implementation of API tests. It is first and foremost a strategy, then a tool selection and then an implementation of more than just API tests. Once set up, and when backed by great analytics, continuous testing improves the quality of applications and entire delivery chains.

Technologies for RESTful API testing

The most common challenge for any type of [automated scripting](#) is locating the right technology for the project. I have found that many teams never get past this phase due to several reasons:

- Unawareness of all the available tools or testing frameworks.
- A lack of the expertise or knowledge needed to customize and deploy an existing framework.
- An inability to afford the cost of vendor tools.

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

- No available time or resources to build a custom testing harness from the bottom up.

Here are some of the tools or frameworks are widely used for automating API testing:

Open Source	Vendor	Custom
<p>Frisby.js + Jasmine</p> <p>Chakram</p> <p>REST-assured</p> <p>Swagger</p> <p>Vows + api-easy</p>	<p>SoapUI Pro</p> <p>Smart Bear</p> <p>Runscope</p>	<p>NodeJS</p>

Conclusion

The testing strategy is not just a piece of paper. It's the reflection of all the activities -- development, testing and the continuous loop of feedback. It improves the entire delivery chain by providing fast feedback and allowing the possibility of iterations during the sprint. And it will allow you to [release your RESTful APIs to the wild](#) with confidence.

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

Testing microservices and APIs in the cloud

Tom Nolle, President, CIMI Corp

*The Internet and cloud computing are quickly shifting application focus from SOA and service busses to **microservices** and **RESTful** design. Application architects and developers find the new models to be more agile, but the very cloud environment that's helping to spawn both trends is making testing microservices and RESTful APIs notoriously difficult.*

To avoid being "trapped in flexibility," developers need to first understand the issues of multidimensional agility the cloud presents, build an approach by applying abstraction principles effectively to reduce complexity, and design services and **APIs** to be flexible in the functional and deployment dimensions without becoming disorderly.

Application agility, meaning the ability of application development to quickly address changes in business needs and opportunities, is increased in large part by componentization and component reuse. Rather than building a totally new application every time something is needed, **componentized development** builds an inventory of reusable components that can be easily assembled and reassembled.

Cloud computing also improves agility by shortening the deployment cycle through the use of public server pools and by facilitating application scaling

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

and component replacement under changes in load or resource status. Beyond that, the cloud is a handy [platform to host microservices](#) or components that are not only broadly useful within a company, but also broadly useful across the market at large.

[Cloud development](#) maturity quickly exposed a critical interdependency here. Testing microservices or APIs in a traditional application can be challenging, but adding a dimension of resource independence and agile component instantiation meant that an application could be tested and certified in one environment and yet fail in what seems a logical evolution of that environment. Quality of experience (QoE) for workers was particularly vulnerable to the interplay of these application agility dimensions.

The [testing process](#) falls victim to too many unconstrained variables and a lack of specific performance or availability objectives for key components of the application. Obviously, the performance of a microservice will depend on where it is hosted in the network relative to the users of the service and how many instances are available to share the load. Yet in most cases the [design of the microservice](#) or API did not include any assumptions on variability in deployment location, number of instances, distribution of work or other things clearly related to cloud deployment.

Behind the [concept of microservices](#), REST and the cloud is the concept of abstraction, which is a technique to expose the features of something without exposing the implementation. A microservice is a functional abstraction, REST abstracts processes to look like simple resources and the cloud creates virtual infrastructure that can map to whatever real resources are available. The challenge for testing microservices [and APIs](#) in the cloud

In this tutorial

- [A roundup of the leading API management tools available today.....](#) p.2

- [What are some solid options for open source API management tools?.....](#) p.14

- [The basics of establishing a RESTful API testing program](#) p.17

- [Testing microservices and APIs in the cloud.....](#) p.24

- [How do I create a secure API for mobile?.....](#) p.29

- [About SearchMicroservices.com](#) p.31

is that the implementation of microservice and REST can collide with cloud resource mapping. Fortunately, that can be fixed if you take care in how your abstractions are created in the first place.

A microservice or API is not a singular web server; that violates the principle of abstracting features from implementation. Thus, microservices and APIs should be viewed as a gateway to an agile collection of implementations below. Experienced cloud developers have said: it's essential to have to view a microservice or RESTful API as a virtual connection that, when implemented, includes agile resource positioning through [DNS](#) or other directory services, load balancing to manage work among instances, and management of collective resources against singular virtual presence. That's exactly correct; the key to testing microservices and APIs in the cloud successfully is the abstraction of the services and APIs to include the resource allocation and component scaling features to be used, both in terms of objectives to be met and practices to be employed. This will enable a test plan to include the cloud-related elements and deal with cloud quality of experience.

Preparing a test plan of this type means including not only the expected load variations, [QoE](#) or scalability requirements, but also what the cloud implementation is expected to do to respond to them. While the number of specific things such a plan has to address varies depending on the application, in general it will be important to do each of the following:

- Identify the specific way in which application component locations are registered on deployment, including the time required.

In this tutorial

- [A roundup of the leading API management tools available today..... p.2](#)
- [What are some solid options for open source API management tools?..... p.14](#)
- [The basics of establishing a RESTful API testing program p.17](#)
- [Testing microservices and APIs in the cloud..... p.24](#)
- [How do I create a secure API for mobile?..... p.29](#)
- [About SearchMicroservices.com p.31](#)

- Specify how work will be distributed among instances of services and ensure that the needed components are also deployed and connected. [Load balancing](#) practices may dictate special attention be paid to state control, though in general REST or microservice APIs are not stateful.
- Identify the means whereby applications or components will detect either a component failure or a need to scale the number of instances and verify these are deployed. It's especially important to set response time goals for this process to set verifiable user-level performance and availability parameters.
- Prepare test data and procedures to drive the collected resource control and functional elements of the application through their expected range of operations, to collect performance data, and validate functionality.

The process of performing the steps above may expose service design or API design issues. The issue of state control has already been noted, but it is possible that [stateless service behavior](#), when applied to horizontally scaled services, could result in issues of transaction recovery or data reliability. This illustrates the importance of designing applications to optimize cloud usage.

Where there might be a profound impact of the cloud on service and [API design](#), it's important to try to capture the specific factors that make this true. This information can then be used to target additional design attention at those applications most likely to benefit from it. Also, record all decisions made for each of the steps above, and be prepared to change your practices if operating experience shows you may have missed important

In this tutorial

- ▶ A roundup of the leading API management tools available today..... p.2

- ▶ What are some solid options for open source API management tools?..... p.14

- ▶ The basics of establishing a RESTful API testing program p.17

- ▶ Testing microservices and APIs in the cloud..... p.24

- ▶ How do I create a secure API for mobile?..... p.29

- ▶ About SearchMicroservices.com p.31

issues. With a little effort, microservice and API design and [testing in the cloud](#) can become routine.

▶ Next article

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

How do I create a secure API for mobile?

Matthew David, Leader, Mobile Center of Excellence, Kimberly Clarke

Security is often an issue that arises when dealing with mobile devices. And as APIs have garnered increased importance with regards to mobile development, the need to [create a secure API](#) for that development becomes important as well. Here we examine the steps needed to create a secure API for mobile.

Inherently, all [mobile devices are insecure](#). The approach to developing mobile apps is to [secure all of the parts at play](#). APIs are an effective way to deliver solutions across multiple platforms -- think of Google Maps, one of the most popular API libraries -- and it is good to develop a secure API for mobile, ensuring it's locked down tight.

There are several steps to achieving a secure API. They include:

- **Data at rest and data in transit.** Your API will move data back and forth across the cloud and to devices. [Leverage HTTPS](#) to protect your data in transit and encrypt the data when at rest on the server and the client.
- **API keys.** Create APIs that require developer registration. The focus for API keys is to lock down and know which apps are using your APIs. The API key is unique to each developer and should be stored on your server in [Base64 encryption](#).

In this tutorial

- A roundup of the leading API management tools available today..... p.2

- What are some solid options for open source API management tools?..... p.14

- The basics of establishing a RESTful API testing program p.17

- Testing microservices and APIs in the cloud..... p.24

- How do I create a secure API for mobile?..... p.29

- About SearchMicroservices.com p.31

- **Oauth2.** [Oauth](#) is a popular authentication format that has been improved with Oauth2, a token based authentication solution that is ideal for securing mobile
- **JWT (JSON Web Token).** Take security of your API over the top through the [inclusion of a JWT](#), a new specification that gives you the tools to create random tokens that can be published to devices, expire at a specific time and can hold [JSON](#) information

The goal is to protect the data on the cloud server as it moves to the API; use tools that [ensure the data is encrypted](#) as it is stored on a device; and, finally, only show data with the correct authentication. This level of security is required for a secure API for mobile. Easy, right?

➤ About SearchMicroservices.com

In this tutorial

- A roundup of the leading API management tools available today..... p.2
- What are some solid options for open source API management tools?..... p.14
- The basics of establishing a RESTful API testing program p.17
- Testing microservices and APIs in the cloud..... p.24
- How do I create a secure API for mobile?..... p.29
- About SearchMicroservices.com p.31

About SearchMicroservices.com

Over 4 million programmers, architects, IT managers, and developers turn to our site for industry news, expert advice and peer-to-peer learning opportunities around managing microservices and service-oriented architecture (SOA), application modernization and digital transformation, Business Process Management (BPM), application integration and APIs, software containers and microservices design and development, DevOps, and more.

For further reading, visit us at: www.SearchMicroservices.com

Images; Fotalia

© 2017 TechTarget. No part of this publication may be transmitted or reproduced in any form or by any means without written permission from the publisher.