

رایانش ابری از منظر معماری سرویس گرا

توسط

فاطمه سلطان محسنی

افشین عباسپور

کلمات کلیدی

معماری سرویس گرا ، رایانش ابری ، وب سرویس ، SaaS ، PaaS ، IaaS ، پردازش سرویس گرا ، SOA ، UDDL ، WSDL و SOC

چکیده

با توجه به رقابت شدید بین موجودیتهای مختلف در دنیای امروز و پیشرفت سریع تکنولوژی های اطلاعاتی و ارتباطاتی مختلف بقا در دنیای امروز مستلزم استفاده درست از داده ها میباشد و رشد داده ها نیز بصورت باور نکردنی در حال افزایش است . بنابر این برای پردازش ، ذخیره سازی و استفاده از این اطلاعات نیاز به راهکارها و روشهای جدیدی است که امکان استفاده از این اطلاعات را در هر زمان و هر مکان و با استفاده از هر نوع دستگاه و ماشین پردازشی و تحت هر نوع پلتفرمی با کمترین هزینه و سریعترین زمان پاسخ گویی میسر نماید . رایانش ابری موضوع نه چندان قدیمی با توانایی ها و فرصتهای به شدت قابل رشد میباشد که شرکتهای بزرگی مانند Google ، Amazon ، IBM و Microsoft به عنوان پیشگامان در این عرصه ظاهر شده و همچنان به توسعه راهکارهای رایانش در محیط های اینترنتی میپردازند .

این مقاله ابتدا به معرفی و بررسی مختصر سرویس گرایی میپردازد و سپس با معرفی رایانش ابری و معماری ها و خدمات مربوطه به بررسی پردازش ابری در تعامل با معماری سرویس گرا و بررسی مشکلات و توانایی های ترکیب این دو میپردازد .

بدون شک برای رسیدن به هدف این مقاله ابتدا نیاز به معرفی مفاهیم خاص و مرتبط با موضوع مقاله میباشد که ابتدا به معرفی مختصر آنها می پردازیم .

| | |
|----|---|
| ۱ | کلمات کلیدی |
| ۱ | چکیده |
| ۳ | سرویس چیست ؟ |
| ۳ | معماری سرویس گرا |
| ۴ | خصوصیات معماری سرویس گرا |
| ۵ | مزایای معماری سرویس گرا برای برنامه های کاربردی |
| ۵ | وب سرویس |
| ۶ | رایانش ابری |
| ۷ | نرم افزار به عنوان سرویس (SaaS) |
| ۷ | بستر به عنوان سرویس (PaaS) |
| ۸ | زیرساخت به عنوان سرویس (IaaS) |
| ۹ | رایانش ابری و رایانش سرویس گرا |
| ۹ | چالشهای مرتبط |
| ۱۰ | در دسترس بودن سرویس |
| ۱۰ | تامین راه حل های امن |
| ۱۰ | نگهداری گسترده تر از جریان کاری |
| ۱۱ | مدیریت مستمر جریانهای کاری سرویسها |
| ۱۱ | فرصتهای دگرگون کننده متقابل |
| ۱۱ | توسعه سریع سرویس |
| ۱۲ | بررسی مقایسه ای SaaS و SOA |
| ۱۲ | نتیجه |
| ۱۳ | منابع |

سرویس چیست ؟

در دنیای واقعی ما همواره در حال استفاده از سرویس های گوناگون هستیم که در واقع این سرویسها خدماتی است که افراد یا بخش های گوناگون به ما ارائه میدهند . اما اگر بخواهیم تعریف مشخصی از سرویس ارائه دهیم باید بگوییم که سرویس وظیفه ای در حال ارائه مستمر است که به طور مشخص و تعریف شده میباشد و بطور نسبی از دیگر وظائف قابل تفکیک میباشد .

اما در دنیای سرویس گرایی^۱ سرویس به صورت یک مفهوم عام نمی باشد و دارای ویژگی های مشخص و معینی است . از جمله این ویژگی ها قابلیت استفاده مجدد ، ارتباط سست و ... میباشد که وظیفه مربوطه را به صورت یک خدمت مستمر و برقرار تعیین میکند .

معماری سرویس گرا

همان طور که اشاره شد معماری سرویس گرا در واقع مجموعه ای از سرویس ها میباشد که با یکدیگر ارتباط دارند و با این ارتباط میتوانند با انتقال داده ها به صورت هماهنگ بین یکدیگر برخی از عملیاتیهای مربوطه را انجام دهند . به عبارتی دیگر معماری سرویس گرا رهیافتی است برای ایجاد سیستمهای توزیع شده که کارکرد نرم افزار را در قالب سرویسهایی ارائه میکند که این سرویسها میتوانند توسط نرم افزار های دیگر و یا حتی سرویسهای دیگر مورد استفاده قرار گیرند . این ترکیب سرویسها و ایجاد سیستمی یکپارچه فرصت ها و توانایی هایی را به سیستم طراحی شده می فزاید که هر روز معماران و طراحان نرم افزار بیشتری را به سوی خود جذب مینمایند . این ارتباطات و تعاملات بین وب سرویسها به دو شکل ایجاد میگردد .

Orchestration (۱)

Choreography (۲)

در سیستم با مدل Orchestration یک واحد اصلی مجموعه ای از سرویسها را فراخوانی میکند و در واقع ترتیب ، ترکیب و تعامل فرآیند های کسب و کار در یک سازمان یا سیستم تحت قوانین و بایدها و نبایدهای مشخص و با نظم واحد اجرا میگردد.

در Choreography تعامل بین چند فرآیند یا سرویس بدون یک رهبر مرکزی صورت میگیرد و سرویسها در کنار هم و بطور هم سطح اجرا میگردند و با همکاری یکدیگر هدفی را محقق میسازند . نمونه ای از این روش را میتوان در پردازش توزیع شده مشاهده کرد .

^۱ SOA Service Oriented Architecture

خصوصیات معماری سرویس گرا

- (۱) برنامه های با معماری سرویس گرا دارای رابط های خود توصیفگر به صورت XML میباشند که وابستگی به سکو^۲ در آن وجود ندارد.
- (۲) سرویسهای معماری سرویسگرا توسط پیامهایی با ساختار مشخص و از پیش تعریف شده با سرویس گیرنده ارتباط برقرار میکنند.
- (۳) سرویسها توسط استاندارد^۳ WSDL قابل تعریف بوده و استفاده کنندگان از سرویس مورد نظر میتوانند به طور واضح و روشن توسط استاندارد^۴ UDDI سرویس مذکور را جهت استفاده ثبت نمایند.
- توضیح: UDDI یک روش و مکانیزم مستقل از سکو است و بر پایه XML بنا شده و مشخص میکند که واحد های مختلف در اینترنت چگونه وب سرویس مورد نظر را ثبت و جستجو نمایند و در واقع مخزنی است که سرویس در آنجا شناخته میشود.
- (۴) سرویسها در معماری سرویسگرا دارای کیفیت سرویس^۵ QOS میباشند که در واقع خط مشی هایی است که معین میکند چه افرادی و چگونه میتوانند سرویس ها را فراخوانی نمایند.
- (۵) سرویسهای مورد نظر را با هر پلتفرمی میتوان ایجاد کرد و در هر پلتفرمی نیز قابل استفاده میباشد.

شباهت هایی بین یک وب سرویس و یک برنامه تحت وب وجود دارد اما این دو بکلی با هم متفاوت میباشند. درواقع تفاوت Web Service و Web Application از نظر کاربر نهایی در این است که برنامه وب توسط کاربر استفاده میگردد اما سرویس وب توسط برنامه ای دیگر مورد استفاده قرار میگیرد. که این برنامه با دانستن استاندارد های مربوط به SOA به راحتی و بدون وابستگی به زبان برنامه نویسی و پلتفرم میتواند از سرویس مورد نظر استفاده نماید. SOA راهکاری است که علاوه بر برنامه های تحت وب میتواند برای برنامه های دیگر نیز مورد استفاده قرار گیرد و استفاده از این روش به سرعت در حال گسترش میباشد.

جدول زیر مقایسه SOA با راهکارهای قبل از به وجود آمدن SOA را مقایسه میکند:

| SOA | رهیافتهای گذشته |
|-------------------------|----------------------------|
| ساخت برای تغییر | ساخت برای بقا |
| توسعه تدریجی | تولید یکباره |
| مستقل از سکو (پلتفرم) | وابسته به سکو ^۲ |
| اتصال سست | اتصال محکم |
| پیام محور | شیء گرا |

^۲ Platform

^۳ Web Service Description Language

^۴ Universal Description Discovery and Integration

^۵ Quality Of Service

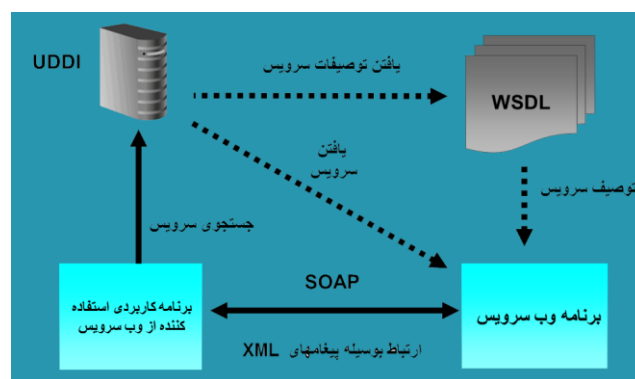
مزایای معماری سرویس گرا برای برنامه های کاربردی

- (۱) استفاده مجدد بهتر و راحت تر
- (۲) ایجاد تغییرات بدون تاثیر روی برنامه های کلاینت (نگهداری راحت تر)
- (۳) انعطاف پذیری بیشتر

وب سرویس

وب سرویس یکی از فعالترین و پراستفاده ترین نمونه های پیاده سازی شده بر اساس معماری سرویس گرا میباشد که با پروتکل SOAP^۶ سازگار میباشد که همه ارتباطات بین سرور و ماشین سرویس گیرنده و سرور ها با هم و یا ماشین های کلاینت با یکدیگر از پروتکل واحد استفاده می کنند که استفاده از این پروتکل واحد باعث شده که به وسیله وب سرویسها امکان ارتباط برنامه ها با زبانهای برنامه نویسی متفاوت و بر روی پلتفرمهای مختلف وجود داشته باشد . وب سرویسها توسط کلاینت ها و به وسیله UDDI شناسایی میگردند که UDDI اطلاعات وب سرویس را در قالب مفهوم استاندارد به نام WSDL منتشر میسازد .

یک WSDL بر طبق استاندارد XML میباشد و فرمت انتقال پیام در وب سرویس را مشخص میکند .



شکل ۱ - معماری وب سرویس

^۶ Access Protocol Simple Object

رایانش ابری

ابتدا به تعریف ابر میپردازیم

ابر، تصویری است انتزاعی از شبکه ای عظیم و توده ای که حجم آن مشخص نمیشود و معلوم نیست که از چه میزان منابع پردازشی تشکیل شده و ابعاد زمانی و مکانی یکایک اجزای آن نیز مشخص و شناخته شده نیست. همچنین نمی دانیم سخت افزار ها و نرم افزارها در کجای این توده قرار دارند، تنها میدانیم که چه چیزی را عرضه میکند.

«پردازش ابری» یک فناوری جدید است که به تازگی از طرف شرکت های بزرگ نرم افزاری مورد توجه قرار گرفته است و برخی از این شرکتها به تجارت و کسب در آمد از این فناوری روی آورده اند و خدمات بسیار مفیدی در این زمینه عرضه شده است و روند رو به رشد پرسرعتی دارد. در این سیستم به جای نصب چندین نرم افزار یا سرویس خاص و حتی امکانات اولیه سخت افزاری و نرم افزاری بر روی چند رایانه، تنها یک نرم افزار، یک بار اجرا و بارگذاری می شود و تمامی افراد از طریق یک خدمات بر خط^۷ به آن دسترسی پیدا می کنند. به این پروسه «پردازش ابری» یا در اصطلاح Cloud Computing می گویند. در واقع پردازش ابری امکان دسترسی آسان و فراگیر شبکه به منابع مشترک و قابل پیکربندی (مانند شبکه، سرور، منابع ذخیره سازی، برنامه ها و سرویس ها) و اختصاص و آزادسازی آنها به سرعت و با کمترین هزینه و راحت ترین روش میباشد. در رایانش ابری برای نرم افزار، سخت افزار یا شبکه پولی پرداخت نمی کنید، بلکه توان محاسباتی و سرویس های نرم افزاری مورد نیازتان را خریداری می کنند. این ایده در واقع صرفه جویی بزرگ و بهره وری زیادی در منابع IT را به همراه خواهد داشت.

از جمله مهمترین و مشهورترین شرکت هایی که این فن آوری را برای کاربران در اختیار میگذارند، میتوان به Google , Yahoo, Amazon, Microsoft, Salesforce اشاره کرد.

ویژگی های کلیدی رایانش ابری پرداخت بر اساس میزان استفاده ، مقیاس پذیری بالا ، خودکارسازی ، خود سرویس دهی از طریق API ، دسترسی از طریق اینترنت ، مجازی سازی و ... میباشد .

خدماتی که رایانش ابری ارائه می دهد شامل ۳ دسته زیر میباشد :

۱. نرم افزار به عنوان سرویس (SaaS)
۲. بستر به عنوان سرویس (PaaS)
۳. زیرساخت به عنوان سرویس (IaaS)

^۷ online services

نرم افزار به عنوان سرویس (SaaS)

در SaaS نرم افزار به عنوان سرویسهایی هستند که به محض درخواست کاربر به وی خدمات میدهند و عملیات مورد نظر را انجام میدهند و اغلب در مرورگرها نمایش داده میشوند. این سرویس کاربران را از مشکلات خرید، نصب و نگهداری نرم افزار مورد نظر روی دستگاه خود بی نیاز میسازد. همچنین کاربر میتواند مطمئن باشد که نرم افزار مورد نظر همیشه و در همه جا قابل دسترس وی می باشد زیرا نرم افزار روی یک یا چند سرور مشخص نصب میگردد و از طریق ابر قابل دسترس میباشد.

برای نمونه هایی از این سرویس میتوان به office۳۶۵، Gmail، Online CRM، Google Docs، Photoshop Online و نمونه های دیگری مانند انواع بازی های آنلاین و وبلاگها که همچنان در حال توسعه میباشد اشاره کرد.



شکل ۲

بستر به عنوان سرویس (PaaS)

به کمک این سرویس مشتری این امکان را دارد که نرم افزار خریداری شده یا ساخته شده توسط خود را، روی بستر ابری قرار دهد و آن را کنترل و تست کرده یا تغییر دهد. PaaS معمولاً به ارائه راهکارهایی برای توسعه نرم افزار توسط سرویسهایی جهت کمک به طراحی، توسعه، تست، انتشار و همچنین مدیریت و میزبانی برنامه روی ابر میپردازد که استفاده از این خدمات اغلب نیاز به نصب و دانلود نرم افزار و یا تیم پشتیبانی متمرکز از جهت جغرافیایی ندارد. به وسیلهی این سرویس دیگر نیازی به خریداری سرویس دهنده ی اختصاصی و گران قیمت نیست و در این حالت تنها محدودیت مشتری عدم نظارت بر زیرساخت زیرین، شبکه یا سرویس دهنده هاست.



شکل ۳

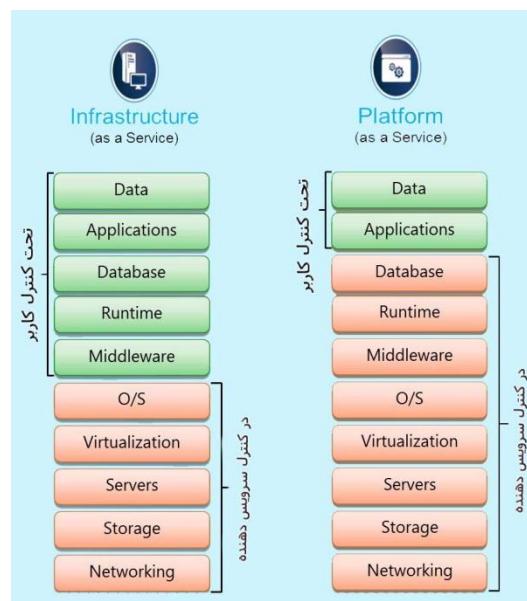
برای نمونه هایی از این سرویس میتوان به Microsoft Azure ، App Engine Google ، Amazon Elastic و EngineYard اشاره کرد .

زیرساخت به عنوان سرویس (IaaS)

این سرویس با فراهم کردن توان پردازشی، شبکه‌ها، فضای ذخیره‌سازی و منابع پایه‌ای رایانشی، نیاز مشتری را به خرید سخت‌افزار و تجهیزات مختلف شبکه از بین می‌برد. این سرویس بر روی لایه دیتا سنتر ایجاد میگردد و توان محاسباتی، ذخیره سازی و ارتباطات شبکه ای را برای مشتری شبیه سازی نماید . کاربران میتوانند این منابع پردازشی را به صورت داینامیک و در صورت درخواست کم یا زیاد نمایند . یکی از نمونه های مطرح این سرویس، خدمات رایانش ابری EC₂ شرکت آمازون است .

از نظر مفهومی و نوع سرویس دهی شباهت هایی بین PaaS و IaaS وجود دارد اما از نظر تحت کنترل بودن برای کاربر IaaS امکانات بیشتری نسبت به PaaS وجود دارد و این یکی از دلایل پرهزینه تر بودن IaaS نسبت به PaaS میباشد .

شکل زیر به بررسی مقایسه ای این دو سرویس از دید کاربر و سرویس دهنده می پردازد .



شکل ۴_ مقایسه Paas و IaaS از نظر دسترسی به منابع

رایانش ابری و رایانش سرویس گرا

رایانش ابری و رایانش سرویس گرا^۸ ارتباطی دوجانبه و متقابل با یکدیگر دارند. یکی پردازش سرویس را تامین میکند و دیگری سرویس های پردازشی را ایجاد می نماید. بهر حال رایانش سرویس گرا در محیط ابری چالشهای تحقیقاتی جدیدی را مطرح مینماید که در ادامه به بررسی هایی در این رابطه میپردازیم.

همان طور که گفته شد سرویسها در معماری سرویس گرا میتوانند بطور جداگانه و یا در ترکیب با سرویسهای دیگر فراخوانی شوند. در نتیجه شرکت ها و سازمانهای مختلف میتوانند سیستمها و نرم افزار های توزیع شده عظیم را با ایجاد و نصب این سرویسهای اولیه بصورت کاملاً پویا طراحی نمایند. از طرفی رقابت بین سازمانها و شرکت ها باعث شده نیاز به تغییر مستمر سیستمهای اطلاعاتی با اضافه کردن ویژگی های جدید و یا حذف موارد قدیمی در زمان کوتاه مشاهده شود.

SOC قصد دارد تا با استفاده از سرویسها به عنوان واحدهای اولیه برای ساخت و ایجاد سریع، کم هزینه و قابل اطمینان سیستمهای نرم افزاری اقدام نماید. البته پردازش سرویس گرا چیز تازه ای نیست و حدود ۳۰ سال پیش توسط شرکت IBM در کامپیوترهای MainFrame در قالب معماری کلاینت / سرور ارائه گردید. به تدریج مدل کلاینت / سرور جای خود را به مدل ابری داد که در آن محدودیتی برای سرویس گیرنده برای اندازه سرور وجود ندارد و داده ها میتوانند بر روی چندین سرور قرار داشته باشند.

در واقع پردازش ابری با افزودن مقیاس پذیری و پردازش شبکه ای تا اندازه زیادی بهبود بخشیده و گسترده تر کرد. همان طور که میدانید همه برنامه های مبتنی بر معماری سرویسگرا لزوماً بر پایه ابر نمی باشند و نیازی نیست تا وب سرویس حتماً بر روی یک میزبان در فضای ابر قرار گرفته باشد. معماری ابری در واقع بهترین راه برای میزبانی یک وب سرویس در شرایطی است که به واقعیتهای زیر نزدیک گردیم:

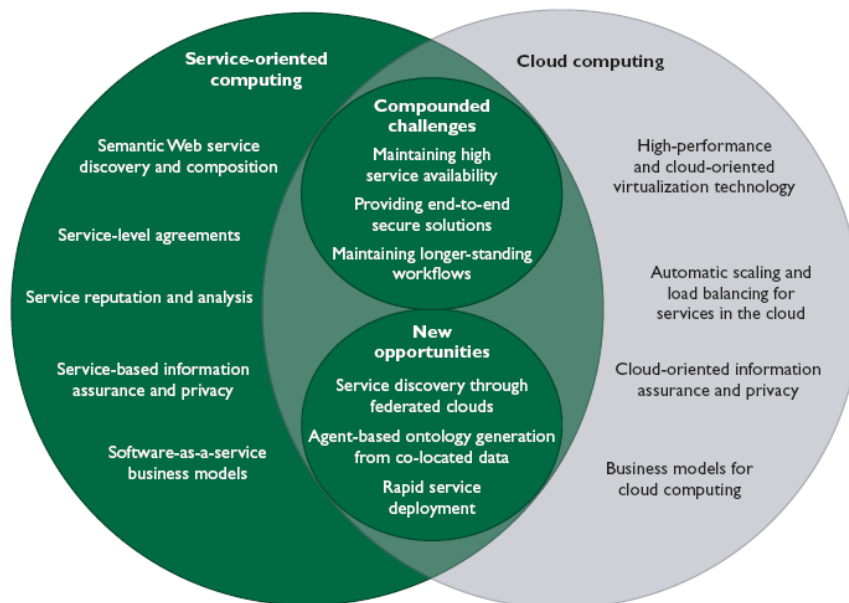
- محدودیتی برای اندازه داده ها وجود نداشته باشد.
- محدودیتی برای توان پردازشی وجود نداشته باشد.
- محدودیتی برای تعداد نمونه های سرویس گیرنده وجود نداشته باشد.

بنابراین بهترین معماری برای SOA و نرم افزار های مبتنی بر آن این است که محدودیتی برای منابع، توان پردازشی و زمان پردازش وجود نداشته باشد، که معماری ابری در کنار پردازش شبکه ای این اطمینان را برای اینکه برنامه های مبتنی بر SOA بهترین قابلیتها برای پردازش سرویس یا کمترین هزینه را داشته باشند حاصل میکند.

چالشهای مرتبط

سیستمهای رایانش سرویس گرا و پردازش ابری دارای ایده ها و و چالشهای نزدیک به هم میباشند و تعجب آور آنکه چالشهای یکی فرصتهایی را برای دیگری مطرح میکند. شکل زیر نشان میدهد که از چالشهای منحصر به فرد SOC و رایانش ابری به وسیله ترکیب این دو مقوله فرصتهایی بدست خواهد آمد.

^۸ SOC Service Oriented Computing



شکل ۵ _ چالشهای مشترک Cloud computing و SOC

در دسترس بودن سرویس

اغلب سیستمهای مبتنی بر SOC نیاز به نگهداری سرویسها و نگهداری برای در دسترس بودن زیاد را دارا میباشند. با این حال رفع این نیازها در محیط پردازش ابری پیچیده تر و مشکل تر میگردد. زیرا سرویسها در زیر ساخت ابر وجود دارند و در نتیجه در دسترس بودن و پاسخ گویی این سرویس ها نه تنها به سرویس بلکه به فراهم کننده ابر نیز وابسته است.

به هر حال خرابی ها اغلب به دلیل قطع برق و خطاهای انسانی میباشد که یک راهکار برای این موضوع استقرار این سرویسها در سرتاسر فراهم کنندگان ابر و بر روی رایانه های متعدد میباشد که البته میتواند در دسترس بودن و فراهم نمودن افزونگی را افزایش دهد.

تامین راه حل های امن

دغدغه حفظ امنیت برای هر دو مقوله رایانش ابری و SOC وجود دارد. در SOC سرویس اصلی تشکیل شده از سرویسهای ترکیبی ممکن است از تامین کنندگان مختلفی در سازمان ایجاد گردد. این موضوع اطمینان از راهکارهای امنیتی نظیر به نظیر را با مشکل مواجه می نماید و ممکن است لازم باشد که همه تامین کنندگان سرویسها، سرویسهای هم سطح خود را تایید و تضمین نمایند و این نیاز زمانی که سرویسها در محیط رایانش ابری وجود دارد بسیار بیشتر میگردد.

نگهداری گسترده تر از جریان کاری

یکی از جنبه های مهم SOC عبور پیام میباشد. به منظور ایجاد جریان کاری بین سرویسها و یا بین سرویسها و نگهدارنده سرویس پیامهایی رد و بدل می گردد. در این رابطه یک مسئله مهم تهدیدات امنیتی و مشکلات حفظ حریم خصوصی و اطلاعات شخصی می باشد. علاوه براین ریسک ردیابی پیامها توسط رقبای امنیتی وجود دارد و ممکن است قادر به استنباط کلی کسب و کارها با استفاده از بررسی ترافیک پیامها باشند.

مدیریت مستمر جریانهای کاری سرویسها

SOC به مدیریت اتصالهای سست برای نگهداری و صحت عملکرد آنها نیازمند است. یک سرویس را میتوان به چندین برنامه کاربردی پیچیده متصل نمود و به این ترتیب قطع یک سرویس میتواند اثرات مخرب متعددی به برنامه های کاربردی وابسته به آن اعمال نماید. در محیط های ابری معضلات و چالشهای مدیریت و نظارت به صورت مشخص تری وجود دارد زیرا ارائه کنندگان کنونی ابر به دلایل متعدد مکامیزم نظارتی بر زیر ساخت های خود را ارائه نمیدهند. از این رو توسعه دهندگان هنوز خود مسئول تامین برنامه و ابزار های نظارت بر سرویسها هستند.

فرصتهای دگرگون کننده متقابل

حال که به بررسی چالشهای رایانش ابری و رایانش سرویس گرا پرداختیم بهتر است به بررسی نقاط قوت این دو مقوله در کنار هم بپردازیم. نقاط قوتی که تا حد بسیار زیادی نقاط ضعف و چالشهای مطرح شده را کمرنگ تر می نماید.

سرویس کشف از طریق اتحاد ابرها:

در عمل یکی از چالشهای بزرگ SOA بحث کشف سرویس بطور خودکار میباشد. فناوری ها و تکنولوژی های موجود برای این امر در اینترنت محدود هستند. همچنین UDDL به خوبی توسعه داده نشده و به اندازه مورد نیاز بطور گسترده به تصویب نرسیده. با این حال در نسل بعدی سرویسهای شروع به توسعه در محیط های ابری کرده اند که این سرویسهای بطور موثری بر این مقوله متمرکز شده اند. به عنوان مثال پایگاه های داده ای توزیع شده پرس و جوی اطلاعات را در سراسر داده های توزیع شده متعدد تسهیل می نماید و به این وسیله ابرها راحت تر میتوانند سرویس مورد نظر را جستجو نمایند. البته برای تحقق بخشیدن به این فرصت جامعه ابر باید اطلاعات مبتنی بر سرویس رامطابق انواع اطلاعات قابل استفاده در سرویس UDDL تغییر دهند و پرتکلهای جدیدی نیز برای اتصال متقابل ابر ها ایجاد گردد تا اطلاعات بتوانند بطور ناشنای برای سراسر تامین کنندگان ابر به اشتراک گذاشته شوند.

توسعه سریع سرویس

SOC این اصل را که وب سرویس ها بتوانند از هر جایی در دسترس باشند و با سرویس های دیگر برای ساخت جریان کاری در سطوح بالاتر ترکیب شوند را دنبال میکند. هرچند که ترکیب و سازمان دهی مجدد و تغییر مکان سرویس های موجود به راحتی امکان پذیر نیست اما در آینده API های استاندارد ابر به تامین کنندگان سرویس اجازه میدهد که سرویس های خود را روی تامین کنندگان پردازش ابری به صورت یکپارچه توسعه دهند. این موضوع به توسعه سریع تر سرویسها کمک خواهد کرد.

بررسی مقایسه ای SaaS و SOA

- SaaS بر روی میزبانی نرم افزار به عنوان سرویس تاکید دارد در حالی که SOA بر روی نرم افزار طراحی شده به عنوان سرویس تاکید دارد .
- SaaS ممکن است به عنوان یک الگوی مصرف مناسب برای کاربر در نظر گرفته شود در حالی که SOA به عنوان مدل طراحی نرم افزار میباشد که در آن هیچ محدودیتی از نظر نوع مصرف کننده وجود ندارد که میتواند هر نوع کاربر و یا سرویس و یا برنامه کاربردی باشد .
- SaaS تا حد زیادی به محیط وب وابسته است اما SOA میتواند در محیطی غیر از وب سرویس خدمات ارائه نماید

بهر حال و گذشته از همه اینها SaaS در واقع بر طبق مفاهیم SOA پیاده سازی میگردد و سرویس های SaaS بر روی ابر از وب سرویسها و عناصر دیگری که از وب سرویسها ساخته شده اند تشکیل میگردند .

نتیجه

امکان پردازش ابری توسط شرکتهای مختلفی ارائه میشود که این ارائه در قالب سرویسهایی در سطوح مختلف و برای افراد مختلف میباشد . برای عموم استفاده کنندگان از ابر نرم افزار در قالب سرویس و برای توسعه دهندگان نرم افزار که نیاز به سکویی جهت توسعه و ارائه نرم افزار در محیط ابر را دارند پلتفرم در قالب سرویس و برای مهندسين شبکه و مدیران سیستمها امکانات زیرساختی و زیر بنایی در قالب سرویس ارائه میگردند که تمامی این خدمات در قالب سرویسهایی با معماری مشخص به کاربر ارائه میگردد . از طرفی معماری مشخص و نه چندان جدیدی برای ارائه سرویسها بصورت مستقل از سکو ، با انعطاف زیاد و با قابلیت مقیاس پذیری وجود دارد که در واقع رایانش ابری میتواند قسمتی از حیطه فعالیت معماری سرویس گرایی به حساب آید .

بررسی راهکارهای معماری سرویس گرا و استفاده از تجربیات آزموده شده در این رابطه بی شک به توسعه رایانش ابری و رفع مشکلات آن کمک شایانی خواهد کرد . ضمن اینکه این تعامل در جهت رشد و توسعه دو جانبه بوده و SOA نیز از فرصتهای پیش روی رایانش ابری بهره مند میگردد . در نتیجه استفاده این دو در کنار هم می تواند ما را به اهداف مشترک آنها نزدیک تر نماید و توانایی های بیشتری را در محیط های ابری اینترنت در اختیار خدمات دهندگان و مصرف کنندگان ارائه دهد .

منابع

- [۱] Wei-Tek Tsai , “**Service-Oriented Cloud Computing Architecture**”, New Generations (ITNG), Seventh International Conference on April ۲۰۱۰
- [۲] Yi Wei , M. Brian Blake , “**Service-Oriented Computing and Cloud Computing**” , Internet Computing paper, IEEE Nov.-Dec. ۲۰۱۰
- [۳] Longji Tang , “**Enterprise Cloud Service Architecture**” , Cloud Computing (CLOUD), ۲۰۱۰ IEEE ۳rd International Conference on July ۲۰۱۰
- [۴] Chauhan, M.A , “**Migrating Service-Oriented System to Cloud Computing: An Experience Report**” Cloud Computing Conference (CLOUD), ۲۰۱۱ IEEE International on July ۲۰۱۱
- [۵] www.ESOA.ir : مقاله کمیته فنی اطلاعات , “**معماری سازمانی سرویسگرا**” , مهندس امیر مهجوریان