یازدهمین کنگره ملی سراسری
فناوریهای نوین در حوزه توسعه پایدار ایران
11th National Congress of
the New Technologies in Sustainable Development of Iran
senaconf.ir

بررسی تکنیک های تحمل پذیری خطا در رایانش ابری

علی پیوندی[۱]، ریحانه مسجودی شچانی[۲]، رزیتا جمیلی اسکویی*[۳]

[۱] دانشجو کارشناسی ارشد،دانشگاه آزاد اسلامی واحد الکترونیکی، تهران Ali.Payvandi@Gmail.com

[۲] دانشجو کارشناسی ارشد،دانشگاه آزاد اسلامی واحد الکترونیکی، تهران Reihane.m@Gmail.com

[۳] استاد راهنما ،دانشگاه آزاد اسلامی واحد الکترونیکی، تهران Rozita2020j@Gmail.com

چکیده

رایانش ابری به عنوان معجزه فناوری اطلاعات قرن بیست و یکم ظهور کرده است که مدل تحویل فناوری اطلاعات را از یک محصول به یک خدمت تغییر داده است. رایانش ابری به دلیل انعطاف پذیری بالا، اقتصادی بودن، قابلیت اطمینان، مدیریت متمرکز، افزایش راندمان و موارد دیگر بسیار محبوب است. رایانش ابری یک سرویس مقیاس‌پذیر را بدون سرمایه‌گذاری اولیه در سرورها، ذخیره‌سازی‌ها یا شبکه‌ها ارائه می‌دهد. محیط پویا ابر به دلیل مقیاسی که در آن کار می کند در برابر خرابی و خطاهای غیرمنتظره بسیار آسیب پذیر می باشد. تحمل خطا (FT) توانایی هر سیستمی برای ادامه عملکرد خود بدون توجه به هر گونه اشکالی (Fault) سخت افزاری یا نرم افزاری غیرمنتظره است. تحمل خطا در رایانش ابری (FTCC) به دلیل پیچیدگی آن یک حوزه تحقیقاتی مهم است. ارائه خدمات بدون اشکال در رایانش ابری برای حفظ اطمینان و رضایت مشتری و جلوگیری از تلفات درآمد ضروری است و تنها در صورتی می توان از خدمات رایانش ابری تا حداکثر پتانسیل خود استفاده کرد که مسائل مربوط به عملکرد مربوط به قابلیت اطمینان، در دسترس بودن و توان عملیاتی به طور موثر توسط ارائه دهندگان خدمات ابری مدیریت شوند و همچنین شکست مورد ارزیابی قراربگیرد بنابراین، تحمل خطا به یک نیاز حیاتی برای دستیابی به عملکرد بالا در رایانش ابری تبدیل می‌شود. در این مقاله در مورد مسائل مربوط به رایانش ابری و تکنیک های تحمل پذیری خطا در آن بحث می کنیم و آنها را مورد مقایسه قرار می دهیم و در پایان نتایج به دست آمده را ارائه می دهیم

واژه‌های کلیدی

تحمل خطا،رایانش ابری،قابلیت اطمینان،خرابی

۱. مقدمه :

رایانش ابری (Cloud computing) مدل رایانشی بر پایه شبکه های بزرگ کامپیوتری مانند اینترنت است که الگویی تازه برای عرضه، مصرف و تحویل سرویس های فناوری اطلاعات (شامل سخت افزار، نرم افزار، اطلاعات و سایر منابع اشتراکی رایانشی) با به کارگیری اینترنت ارائه می کند.

رایانش ابری به دسترسی، پیکربندی و دستکاری منابع (مانند نرم افزار و سخت افزار) در یک مکان راه دور اشاره دارد. رایانش ابری را برحسب محاسبات توزیع شده تعریف می کند: «ابر نوعی سیستم موازی و توزیع شده است که شامل مجموعه ای از رایانه های به هم پیوسته و مجازی شده است که به صورت پویا تهیه شده و به عنوان یک یا چند منبع محاسباتی یکپارچه بر اساس توافق نامه های سطح خدمات ایجاد شده است .مذاکره بین ارائه دهنده خدمات و مصرف کنندگان[1].

با پیشرفت فناوری اطلاعات نیاز به انجام کارهای محاسباتی در همه جا و همه زمان به وجود آمده است. همچنین نیاز به این است که افراد بتوانند کارهای محاسباتی سنگین خود را بدون داشتن سخت افزار ها و نرم افزار های گران از طریق خدماتی انجام دهند. رایانش ابری آخرین پاسخ فناوری به این نیاز ها است. موسسه ملی فناوری و استاندارد ها (NIST) رایانش ابری را اینگونه تعریف می کند: رایانش ابری مدلی است برای فراهم کردن دسترسی آسان بر اساس تقاضای کاربر از طریق شبکه به مجموعه ای از منابع رایانشی قابل تغییر و پیکربندی (مثل: شبکه ها، سرورها، فضای ذخیره سازی، برنامه های کاربردی و سرویسها) که این دسترسی بتواند با کمترین نیاز به مدیریت منابع و یا نیاز به دخالت مستقیم فراهم کننده سرویس به سرعت فراهم شده یا آزاد گردد[3].

رایانش ابری بر پایه مجازی سازی، محاسبات توزیع شده، محاسبات ابزار، و اخیراً شبکه، وب و خدمات نرم افزاری ساخته شده است.

افراد و سازمان‌ها از سخت‌افزار و نرم‌افزار مدیریت شده توسط اشخاص ثالث در مکان‌های راه دور استفاده می‌کنند .ذخیره‌سازی فایل آنلاین، سایت‌های شبکه‌های اجتماعی، ایمیل‌های اینترنتی و برنامه‌های تجاری آنلاین برخی از سرویس‌های ابری رایج هستند کاربر می تواند از این خدمات بدون اطلاع از جزئیات سخت افزاری و نرم افزاری استفاده کند[5].

در حال حاضر، رایانش ابری نقش مهمی در ارائه خدمات محاسباتی به مشاغل با اندازه های مختلف ایفا می کند. خرابی سرویس های ابری می تواند زیان مالی قابل توجهی داشته باشد. در مورد سیستم های بلادرنگ حیاتی، حتی می تواند فاجعه آمیز باشد. برای اطمینان از قابلیت اطمینان بیشتر، سیستم‌های رایانش ابری باید به طور استثنایی نسبت به خطا مقاوم باشند .تحمل خطا (FT) توانایی یک سیستم برای ارائه خدمات مورد نظر حتی زمانی که خرابی ها و خطاها در سیستم اتفاق می افتد، می باشد. رایانش ابری دارای مزایای بسیاری از جمله مقیاس پذیری و انعطاف پذیری سرویس است .با این حال، رایج ترین نگرانی از دیدگاه کاربر، عملکرد، قابلیت اطمینان، کنترل، امنیت و حریم خصوصی است. کاربران ابری انتظار دارند که سرویس ها همیشه در دسترس باشند و از هر مکانی به داده های خود دسترسی داشته باشند[2] .

تحمل خطا با تمام تکنیک‌های اجتناب‌ناپذیر برای فعال کردن استحکام و قابلیت اطمینان همراه است. مزایای اصلی پیاده‌سازی تحمل خطا در رایانش ابری شامل بازیابی شکست، هزینه کمتر، معیارهای بهبود عملکرد است[7].

بخش های این مقاله به صورت زیر مشخص می گردد : پیشینه تحقیق، مفاهیم رایانش ابری،تحمل اشکال (تعاریف،رویکردهاابزارها و مدل ها)و چالش و نتیجه گیری

۱.پیشینه تحقیق :

رایانش ابری زمینه ارائه منابع و خدمات محاسباتی مجازی را در یک محیط مشترک و مقیاس‌پذیر از طریق شبکه بر روی یک مدل پرداخت به‌موقع ارائه می‌کند .با پذیرش سریع محاسبات ابری، بخش بزرگی از شرکت‌های فناوری اطلاعات و سازمان‌های دولتی در

سراسر جهان، خدمات ابری را برای اهداف مختلف از جمله میزبانی برنامه‌های کاربردی حیاتی و در نتیجه داده‌های حیاتی، پذیرفته‌اند .به منظور پشتیبانی از این برنامه‌ها و داده‌های حیاتی، نیاز به فراهم کردن محیط‌های محاسبات ابری قابل اعتماد وجود دارد[8].

پنج کاراکتر اصلی و نقش های مرتبط در محیط های ابری در سند نقشه راه استانداردهای محاسبات ابری NIST توضیح داده شده است. جدول ۱ مصرف کنندگان و ارائه دهندگان دو نقش اصلی در میان این بازیگران هستند[9].

جدول ۱. کاراکترها در رایانش ابری[9]

کاراکتر	تعریف
مصرف کننده ابر	هر فرد یا سازمانی که با ارائه دهندگان ابری رابطه تجاری دارد و خدمات موجود را مصرف می کند
ارائه دهنده ابر	هر نهاد یا سازمان فردی که مسئول ارائه خدمات و ارائه منابع محاسباتی به مصرف کنندگان ابری
کارگزار ابری	یک نهاد IT که ورودی برای مدیریت عملکرد و QoS محاسبات ابری ارائه می کند خدمات علاوه بر این به ارائه دهندگان و مصرف کنندگانابر در مدیریت مذاکرات خدمات کمک می کند
ممیز ابری	گروهی که می تواند ارزیابی مستقلی از خدمات ابری ارائه شده توسط ارائه دهندگان ابر از نظر عملکرد، امنیت و حریم خصوصی، عملیات سیستم اطلاعات و غیره در محیط های ابری ارائه دهد.
حامل ابری	یک طرف واسطه که دسترسی و اتصال را برای مصرف کنندگان از طریق هر وسیله دسترسی مانند شبکه فراهم می کند .حامل ابر خدمات را از یک ارائه دهنده ابر به مصرف کنندگان ابر منتقل می کند

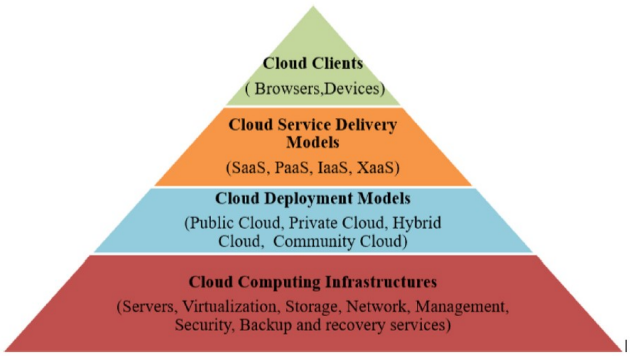
تجزیه و تحلیل نشان می دهد که سه مورد نیاز حیاتی "در دسترس بودن"،"قابلیت اطمینان"و"عملکرد" برای مصرف کنندگان و ارائه دهندگان ابر وجود داردبنابراین از یک سو از دیدگاه مصرف کنندگان ، نیاز ضروری برای در دسترس و قابل اعتماد بودن خدمات ابری وجود دارد اما از سوی دیگر ،در حالی که ارائه دهندگان ابری لزوم ارائه خدمات بسیار در دسترس و قابل اعتماد برای برآورده کردن کیفیت خدمات (QoS) را به دلیل SLA درک می کنند، ولی ترجیح می دهند برای دستیابی به سود بیشتر، سیستم های با منفعت بیشتر داشته باشند[4]. تحت این ملاحظات، ارائه محیط های ابری قابل اعتماد که بتواند خواسته های مصرف کنندگان و ارائه دهندگان ابر را برآورده کند، یک چالش جدید است.

در این مقاله تعاریف مربوط به اشکال،خطا و خرابی بطور عام و ارتباط آنها با یکدیگر و در ادامه تکنیک های تحمل اشکال در رایانش ابری مورد بررسی قرار گرفته است[1].تکنیک های FT در رایانش ابری در دو گروه "تحمل اشکال واکنشی" و "تحمل اشکال فعال(پیشگیرانه)" طبقه بندی شده اند،[2] .

۲.محاسبات ابری

رایانش ابری از تلاش در زمینه های تحقیقاتی مختلف محاسبات مانند محاسبات توزیع شده، محاسبات شبکه ای، تکنیک های مجازی سازی وSOA(معماری سرویس گرا) تکامل یافته است. بنابراین ویژگی ها، پیشرفت ها و محدودیت های آن ها را نیز در خود جای داده است[1].

این بخش محاسبات ابری رادر پنج بعد توصیف می کند : (الف)مفاهیم اساسی (ب)اجزای ابر (ج)زیرساخت ابر (د)مدل استقرار ابر (ه)پشته خدمات ابری، همانطور که در تصویر نشان داده شده است.شکل ۱.



شکل ۱

۱-۲.زیرساخت رایانش ابری :

زیرساخت محاسبات ابری بطور گسترده شامل موارد زیر است :

- سرورها : ماشین های فیزیکی که بعنوان میزبان برای یک یا چند ماشین مجازی عمل می کنند.
- مجازی سازی : فن آوری که اجزای فیزیکی مانند سرورها،ذخیره شازی و شبکه را انتزاع می کندو اینها را بعنوان منابع منطقی ارائه می کند.
- ذخیره سازی : در قالب شبکه های فضای ذخیره سازی(SAN)،ذخیره سازی متصل به شبکه(NAS)،درایوهای دیسک وغیره به همراه امکاناتی مانند آرشیو و پشتیبان گیری.
- شبکه : برای ایجاد ارتباط متقابل بین سرورهای فیزیکی و ذخیره سازی.
- مدیریت : نرم افزارهای مختلفبرای پیکربندی ، مدیریت و نظارت بر زیرساخت های ابری شامل سرورها،شبکه و دستگاه های ذخیره سازی.
- امنیت : اجزایی که یکپارچگی، در دسترس بودن و محرمانه بودن داده ها و امنیت اطلاعات را بطور کامل فراهم می کنند.
- خدمات پشتیبانی گیری و بازیابی.

۲-۲.مدل های استقرار ابری

مدل استقرار ابر بر اساس انگیزه و محیطی است که انتظار می رود یک سرویس ابری در آن استفاده شود. انتخاب مدل استقرار هزینه های متحمل شده، مصرف نیرو توسط منابع و سایر هزینه های سرمایه را تعیین می کند. رایج‌ترین مدل‌های استقرار در محیط‌های ابری، ابر عمومی، ابر خصوصی، ابر جامعه و ابر ترکیبی هستند.

ابر عمومی: ابر عمومی به عموم مردم اجازه دسترسی به سیستم ها و خدمات ارائه شده توسط یک ارائه دهنده سازمانی را می دهد .این انعطاف پذیری، مقیاس پذیری، استقلال مکان را با هزینه بسیار کم فراهم می کند زیرا به طور کلی از چند اجاره ای استفاده می

شود.منابع به صورت پویا و بر اساس درخواست از یک ارائه دهنده شخص ثالث راه دور که منابع را با استفاده از رویکرد چند مستاجر ارائه می کند، تهیه می شود[10].

ابرخصوصی :ابر خصوصی در یک سازمان خاص استفاده می شود، به عنوان مثال، می توان به منابع و خدمات ابری در داخل یک سازمان دسترسی داشت یا از آنها استفاده کرد .این مدل، امنیت و حریم خصوصی داده ها و برنامه های کاربردی بالا را تضمین می کند[10].

ابر جامعه : این مدل توسط شرکت ها یا سازمان های مختلف به طور همزمان مورد استفاده قرار می گیرد و به یک اجتماع یا جامعه خاص که شامل مشارکت های جمعی است (به عنوان مثال ضرورت های امنیتی، ماموریت، و ملاحظات انطباق و غیره) کمک می کند .این مدل ممکن است توسط یک یا چند سازمان در داخل جامعه و یا شخص ثالث اداره و مدیریت شود[10].

ابرهیبریدی :ابر ترکیبی از ابر عمومی و ابر خصوصی است .در این استقرار ابر، رویدادهای حیاتی(مثلا مواردی که به عملیات ایمن نیاز دارند)با استفاده از خدمات ابر خصوصی و رویدادهای غیر بحرانی با استفاده از ابر عمومی اجرا می شوند[11].

۳-۲.مدل های سرویس ابری

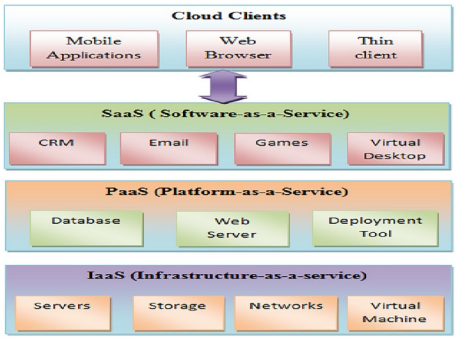
اگرچه رایانش ابری در سال‌های اخیر بسیار تکامل یافته است، خدمات هنوز در سه مدل خدمات اصلی قرار دارند مدل های خدمات اولیه در شکل ۲ نشان داده شده است.

نرم افزار به عنوان سرویس(SaaS) : در این مدل، نرم افزارهای کاربردی توسط ارائه دهنده خدمات ابری در قالب خدمات به مصرف کننده/کاربران نهایی ارائه می شوند[10]. برنامه ای که به عنوان یک سرویس به مشتری ارائه می شود، نیاز به نصب و اجرای برنامه ابری را بر روی رایانه کاربر از بین می برد و تعمیر و نگهداری را ساده می کند. به عنوان مثال، خدمات کنفرانس وب، برنامه های ایمیل، پلت فرم های رسانه های اجتماعی و غیره. لیست ارائه دهندگان Saas عبارتند از: Citrix، Qstack، Enterprise، CloudStack ، IBM ، Microsoft Azure، Google Compute Engine،AWS، CloudForge، Open-Nebula،OpenStack Amazon،SmartCloud و غیره.

پلت فرم به عنوان یک سرویس(PaaS) : این مدل بستری را برای توسعه، اجرا، آزمایش و مدیریت اپلیکیشن ها در فضای ابری فراهم می کند. یک کاربر می تواند یک محیط را با یک پشته نرم افزار از یک CSP اجاره کند و از آن برای توسعه برنامه های کاربردی سفارشی استفاده کند .لیست ارائه دهندگان PaaS عبارتند ازAcquia Cloud ، Amazon AWS، App Agile، Apprenda، AppScale، Bluemix، Cloud 66، Cloud waysو غیره[12].

زیرساخت به عنوان یک سرویس(IaaS) : مدل IaaS امکان دسترسی به برخی منابع اولیه مانند ماشین‌های فیزیکی، ذخیره‌سازی، شبکه‌ها، سرورها، ماشین‌های مجازی روی ابر و غیره را فراهم می‌کند. ارائه‌دهنده IaaS خدماتی مانند تهیه ماشین مجازی پویا و امکانات ذخیره‌سازی بر اساس تقاضا را ارائه می‌دهد. لیست ارائه دهندگان Iaas عبارتند از Salesforce، Microsoft، Amazon Web Services، Slack، Zendesk، GitHub، Oracle، Ciscoو غیره

هر چیزی به عنوان یک سرویس (XaaS) : یکی دیگر از مدل های سرویس است که ممکن است هر چیزی یا همه چیز به عنوان یک سرویس باشد سیستم ابری قادر است حجم عظیمی از منابع را برای برآوردن نیازهای شخصی و خاص با استفاده از امنیت به عنوان سرویس، هویت به عنوان سرویس، ارتباط به عنوان یک سرویس، DaaS(پایگاه داده به عنوان سرویس) یا استراتژی به عنوان یک سرویس و غیره حفظ کند.



شکل ۲

۳.مفاهیم تحمل اشکال

محاسبات تحمل پذیر اشکال یک اصطلاح عمومی است که تکنیک های طراحی اضافی را با اجزای تکراری یا محاسبات مکرر توصیف می کند که عملیات بدون وقفه (تحمل کننده) را در پاسخ به خرابی مؤلفه (عیب ها) ممکن می سازد. کاربردهای زیادی وجود دارد که در آنها قابلیت اطمینان سیستم کلی باید بسیار بیشتر از قابلیت اطمینان اجزای جداگانه آن باشد .در چنین مواردی، طراحان مکانیسم ها و معماری هایی را ابداع می کنند که به سیستم اجازه می دهد تا اثرات اشکال یک جزء را کاملا بپوشاند یا به سرعت آن را بازیابی کند تا برنامه به طور جدی تحت تأثیر قرار نگیرد[15]

تحمل اشکال برای یک سیستم بسیار مهم است تا به آن اجازه دهد خدمات مورد نیاز را حتی در صورت وجود اشکال قطعات یا یک یا چند اشکال ارائه دهد [1].

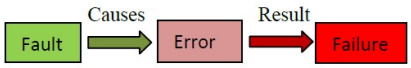
۱-۳.طبقه بندی اشکال(از نظر ماهیت)

بر اساس ماهیت اشکال، می توانیم آنها را به عناوین زیر دسته بندی کنیم[3].

- یک اشکال می تواند نتیجه برخی شرایط موقتی باشد که در سیستم رخ می دهد که ممکن است سیستم را به طور موقت متوقف کند مانند مشکل اتصال و مشکل خدمات غیرعملیاتی در یک شبکه .ما این اشکالات را اشکال های گذرا می نامیم[6].
- یک اشکال می تواند ویژگی های نمایشگاهی را نشان دهد، مانند اینکه ممکن است به طور تصادفی در فواصل زمانی منظم رخ دهد .هیچ تشخیص یا رفع دائمی برای چنین اشکالاتی وجود ندارد .ما آنها را به عنوان اشکال های متناوب خطاب می کنیم[6].
- اشکالی که معمولاً تا زمانی که علت اصلی آن برطرف نشده باشد، وجود دارد .آنها به عنوان اشکال های دائمی خطاب می شوند[6].
- گاهی اوقات تشخیص وجود اشکال در یک سیستم بسیار دشوار است .این اشکال ها بیشترین هزینه را برای حذف و تشخیص آن دارند .ما آنها را به عنوان اشکال های بیزانسی خطاب می کنیم[6].

۲-۳.اشکال،خطا و خرابی

خرابی ها Failure در یک سیستم در نتیجه خطاهایی Error رخ می دهد که به نوبه خود ناشی از اشکال Fault است.شکل ۳[16].

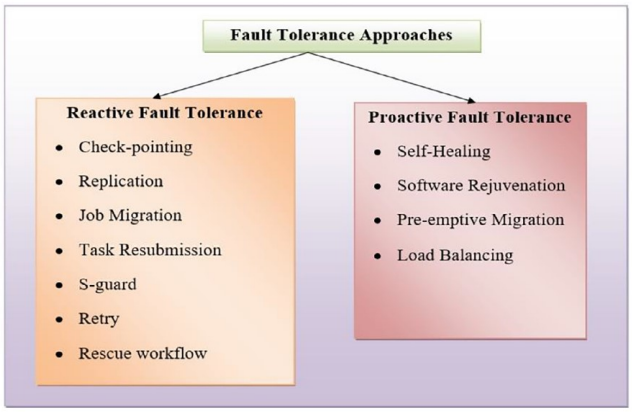


شکل ۳-وقوع خرابی

۴.رویکردهای تحمل اشکال

چندین تکنیک برای پیاده سازی تحمل اشکال در رایانش ابری استفاده می شود .با توجه به نحوه برخورد آنها با اشکال(یعنی قبل و یا بعد از وقوع اشکال)،رویکردهای تحمل اشکال به دو دسته اصلی طبقه بندی می شوند(شکل ۴)[2] .که عبارتند از :

- رویکرد تحمل اشکال واکنشی(RFT)
- رویکرد تحمل اشکال پیشگیرانه(PFT)



شکل ٤ -طبقه بندی رویکردهای تحمل اشکال

۴-۱.رویکرد تحمل اشکال واکنشی(RFT)

تکنیکهای تحمل اشکال واکنشی برای کاهش تأثیر خرابی ها بر روی یک سیستم در زمانی که خرابی ها واقعا رخ داده اند استفاده می شود[5]. این تکنیک ها تلاش می کنند تا خدمات را از طریق تکنیک بازیابی حفظ کنند برخی از این تکنیک ها که به عنوان سیاست واکنشی طبقه بندی می شوند در زیر آورده شده است[2] .

- ایست بازرسی/راه اندازی مجدد (Check-pointing/Restart) : این یک تکنیک تحمل خطا در سطح کار کارآمد برای اجرای طولانی و برنامه های کاربردی بزرگ است. در این سناریو پس از انجام هر تغییر در سیستم، یک علامت چک انجام می شود . هنگامی که یک کار با شکست مواجه می شود، به جای اینکه کار از ابتدا شروع شود مجاز است آن کار را از حالت علامت گذاری شده اخیر مجدداً راه اندازی کند[7].
- مهاجرت شغلی (Job Migration) :اگرکاری به دلایلی نتواند اجرای خود را بر روی یک ماشین فیزیکی خاص کامل کند و با شکست مواجه شود، به ماشین دیگری منتقل می شود.[1] برخی از الگوریتم‌ها وجود دارند که به طور مکرر خرابی‌ها را تعیین می‌کنند و برنامه‌های دسته‌ای را در ابری از چندین مرکز داده منتقل می‌کنند .این کار از طریق HAProxy انجام می شود[14].

همانند سازی (Replication) : برایایجاد کپی های متعدد از وظایف و ذخیره کپی ها در مکان های مختلف استفاده می شود .یک کار می تواند در صورت وجود نقص یا خرابی اجرا را ادامه دهد تا زمانی که همه کپی ها از بین بروند[1]. با استفاده از ابزارهایی مانندHA-Proxy ، Hadoopو AmazonEc2 Replication می توان پیاده سازی کرد[7].

- S-Guard: در این تکنیک در صورت بروز خرابی،کار برگشت داده می شود، S-Guard را می توان در Amazon EC2 و HADOOPپیاده سازی کرد[14] .
- سعی مجددRetry در این رویکرد، یک کار به طور مکرر اجرا می شود تا زمانی که موفق شود .از همین منبع برای آزمایش مجدد کار ناموفق/شکست استفاده می شود[1].
- ارسال مجدد کار Task Resubmission : :کار ناموفق دوباره یا به همان ماشینی که روی آن کار می کرد یا به ماشین دیگری ارسال می شود[5].
- گردش کار نجات (Rescue workflow) : به سیستم اجازه می دهد تا پس از شکست هر کاری به کار خود ادامه دهد تا زمانی که قادر به ادامه کار بدون رفع عیب نباشد[5].
- چک کیسه های ایمنی (Safety-bag checks) : در این حالت مسدود کردن دستوراتی انجام می شود که دارای ویژگی های ایمنی نیستند[17].

۲-۴.رویکرد تحمل اشکال پیشگیرانه(PFT)

تکنیک های تحمل اشکال پیشگیرانه اشکال ها را پیش‌بینی می‌کند و اجزای مشکوک را با سایر اجزای کار جایگزین می‌کند و بنابراین از بازیابی عیب و خطا جلوگیری می‌کند[5]. مروری بر این تکنیک ها در ادامه شرح داده شده است:

- خود درمانی (Self-Healing) : این روش از تکنیک تقسیم و غلبه استفاده می کند، که در آن یک کار بزرگ به چند تکه تجزیه می شود .این تقسیم بندی عمدتا برای بهبود عملکرد سیستم انجام می شود .هنگامی که نمونه‌های متعددی از یک برنامه مشابه روی ماشین‌های مجازی (ماشین‌های مجازی) مختلف اجرا می‌شوند، آنگاه خرابی نمونه‌های برنامه به‌طور خودکار رسیدگی می‌شود .این امکان را به دستگاه ها یا سیستم های محاسباتی می دهد تا هویت خود را شناسایی کنند، معضلات/مشکلات پیش آمده را بدون وابستگی به مدیر تشخیص دهد و آن ها را برطرف کند[1].
- جوان سازی نرم افزار (Software Rejuvenation) : در این رویکرد، سیستم تحت راه اندازی مجدد دوره ای قرار می گیرد و هر بار از یک حالت جدید شروع می شود[1]
- مهاجرت پیشگیرانه (Pre-emptive Migration) : در این تکنیک یک برنامه به طور مداوم مشاهده و تجزیه و تحلیل می شود .مهاجرت پیشگیرانه یک کار، به مکانیسم کنترل حلقه بازخورد بستگی دارد[7][5].
- تعادل بار (Load Balancing) : این رویکرد برای متعادل کردن بار حافظه و CPU در زمانی که از حد حداکثر/مشخصی فراتر می رود استفاده می شود .بار CPU بیش از حد به CPU دیگری منتقل می شود که از حداکثر حد مجاز خود تجاوز نمی کند[1].

شایان ذکر است در برخی مراجع این دسته بندی تکنیک های تحمل اشکال شامل سه گروه می باشد که گروه سوم به نام "روش تحمل اشکال تطبیقی" می باشد در روش تحمل خطا تطبیقی، همه رویه‌ها با توجه به شرایط به صورت خودکار انجام می‌شوند .این اطمینان را می دهد که قابلیت اطمینان ماژول های حیاتی و قابلیت اطمینان ماشین های مجازی پس از هر چرخه محاسباتی تغییر می کند.

در روش تحمل خطا تطبیقی، تکنیک‌های روش‌های تحمل خطای فعال و واکنشی را می‌توان برای رفع عیوب ترکیب کرد .برخی از این تکنیک‌ها عبارتند از ایست بازرسی،همانند سازی،تکه تکه کردن(Fragmentation) و غیره

Fragmentation تکنیکی است که در آن یک فایل به چندین فایل تقسیم می شود به گونه ای که اگر این فایل ها با هم ترکیب شوند بدون از دست دادن داده ها، فایل اصلی را تشکیل می دهد .این تکنیک حجم داده های غیر مرتبط بازیابی شده توسط برنامه ها را کاهش می دهد[14].

۳-۴.معیارهای مورد استفاده برای تحمل خطا در رایانش ابری

رویکردهای تحمل خطا در رایانش ابری با استفاده از پارامترهای مختلف برای بررسی کارایی و اثربخشی .سیستم های ابری ارزیابی می شوند .پارامترهای ممکن در لیست ذکر شده است [7][1].جدول۲

جدول۲: معیارهای تحمل خطا در رایانش ابری[1][7]

پارامترها / معیارها	شرح
توافقی	تمامی فرآیندها به صورت خودکار طبق شرایط اجرا می شوند
کارایی	عملکرد برای اطمینان از کارایی سیستم مورد استفاده قرار می گیرد
زمان پاسخ	کل زمان صرف شده برای پاسخ به یک الگوریتم خاص
توان عملیاتی	تعداد وظایفی را محاسبه می کند که اجرای آنها با موفقیت انجام شده است
قابلیت اطمینان	انگیزه اصلی آن ارائه نتیجه دقیق یا قابل قبول در یک دوره زمانی معین است
دسترسی	به عنوان احتمال توصیف می شود.سیستم در نظر گرفته شده برای استفاده پس از درخواست بدرستی کار کند
قابلیت استفاده	کاربر می تواند از یک اختراع/محصول برای دستیابی به هدف با کارایی، اثربخشی و رضایت استفاده کند.
سربار مرتبط	مقدار سربار مربوط به اجرای الگوریتم تحمل خطا را تعیین می کند .این مقدار از سربار به دلیل جابجایی وظایف، بین پردازنده و ارتباطات بین فرآیندی تشکیل شده است. باید به حداقل برسد تا یک تکنیک تحمل خطا بتواند به طور کارآمد کار کند
مقرون به صرفه بودن	این یک توصیف از سیستم پولی است
مقیاس پذیری	این توانایی یک الگوریتم برای انجام تحمل خطا برای یک سیستم با تعداد محدود گره است .این معیار باید بهبود یابد.

۴-۴.ابزارهای مورد استفاده برای اجرای تحمل خطا

ASSURE :مخفف Automatic Software Self-Healing Using Rescue Points است .این برای نقاط نجاتی کار می کند که شکست های مورد انتظار برنامه نویس را کنترل می کند هنگامی که یک خطا در یک مکان خاص در برنامه رخ می دهد، ASSUREاجرا را به نزدیک ترین نقطه نجات باز می گرداند و برنامه را متقاعد می کند تا اجرا را با مجازی سازی بازیابی کند[14].

HAProxy :مخفف High Availability Proxy است .هدف این ابزار افزایش کارایی و سازگاری یک خوشه سرور با توزیع بار از طریق سرورهای مختلف است .این یک برنامه متن باز است که برای راه حل های متعادل کننده بار برای وب سایت ها توسعه یافته است .از نظر استفاده از پردازنده و حافظه عملکرد خوبی از خود نشان می دهد .همچنین به دلیل پایداری و قابلیت اطمینان آن شناخته شده است . وب سایت هایی مانندTwitter ، Stack Overflow، GitHubو غیره از این ابزار برای غلبه بر مشکل ترافیک سنگین خود استفاده می کنند[14].

SHelp :، یک ترتیب زمان اجرا است که برای ماشین های مجازی کار می کند و نقاط نجات و تکنیک های مجازی سازی خطا را به کار می گیرد .افزونگی را کاهش می‌دهد و به سرعت از خطاهای آینده تحت تأثیر باگ‌های مشابه بازیابی می‌کند SHelp .می تواند برنامه های کاربردی سرور را برای بازیابی از این اشکالات در چند ثانیه ایجاد کند[14].

AmazonEC2(Amazon Elastic Compute Cloud):یک سرویس وب محبوب است که بسته به نیاز کاربر فضایی را در ابر فراهم می کند .این در رسیدگی به نمونه ها از نظر زمان کارآمد است، ابزارهایی را در اختیار توسعه دهندگان قرار می دهد تا برنامه هایی را طراحی کنند که مقاوم در برابر شکست هستند[14].

Hadoop :برخی از خدمات رایانش ابری را ارائه می دهد .این یک پروژه بنیاد نرم‌افزار آپاچی منبع باز است که امکان پردازش توزیع‌شده مجموعه‌های داده عظیم را در میان دسته‌ای از گره‌ها فراهم می‌کند .این خدمات به ابرهای عمومی و خصوصی مانندIaaS ، PaaSوSaaS ارائه می دهد[14].

جدول ۳-مقایسه تکنیک های مختلف تحمل خطا

تکنیک	رسته	ابزار	سربار	هزینه	کارایی	خطای شناسایی شده
Checkpoint	Reactive	S Help ASSURE HAProxy	متوسط	بالا	بالا	برنامه،میزبان،شبکه
Replication	Reactive	Hadoop AmazonEC2	کم	عظیم	بالا	فرایند،گره،برنامه
Migration	Reactive proactive	HAProxy Hadoop	کم	متنوع	متوسط	فرایند،گره،برنامه
Retry	Reactive	ASSURE	بالا	ندارد	متوسط	میزبان،شبکه
S-Guard	Reactive	Hadoop Amazon-EC2	بالا	پائین	متوسط	گره،برنامه

Task Resubmission	Reactive	Amazon-EC2	بالا	پائین	متوسط	گره،برنامه
Self-Healing	Proactive	HAProxy ASSURE	متوسط	بالا	بالا	فرایند،گره،برنامه
Software Rejuvenation	Proactive	Eucalyptus Amazon EC2	بالا	بالا	بالا	فرایند،برنامه
Rescue Workflow	Reactive	Hadoop	پائین	پائین	متوسط	گره،برنامه

۵.نتیجه گیری و چالشهای موجود

۵-۱. چالشهای موجود

چالش های مدل های مختلف تحمل خطا در طول بررسی ادبیات، چالش‌های مختلفی در تحمل خطا در رایانش ابری با چالش‌های زیر مواجه است که عبارتند از

۱- احتمال خطا بسیار زیاد است زیرا پردازش از طریق رایانه های راه دور انجام می شود

۲- نیاز به پیاده سازی روش های کارآمد برای یافتن خطاها وجود دارد

۳- داده های محدود به مشتریان به دلیل تراکم سیستم بالا داده می شود

۴- روش‌های پیش‌بینی خطا برای کاربردهای بلادرنگ بیشتر باید توسعه یابند

۵- درک وضعیت تغییر سیستم دشوار است زیرا محیط ابری فعال است

۶- وقوع خطا در مراکز داده دغدغه سازمان مشتری برای اجرای تکنیک های تحمل خطا نیست.

۵-۲.نتیجه گیری

سیستم ابری مستعد خرابی است. شکست در محاسبات ابری به دلیل مقیاس عملیات اجتناب ناپذیر است. سیاست‌های تحمل خطا معمولاً برای رسیدگی مؤثر به خطاها در محیط ابری اجرا می‌شوند. تکنیک های تحمل خطا به پیشگیری و همچنین تحمل خطاها در سیستم کمک می کند، که ممکن است به دلیل خرابی سخت افزار یا نرم افزار رخ دهد. انگیزه اصلی به کارگیری تکنیک های تحمل خطا در رایانش ابری دستیابی به بازیابی شکست، قابلیت اطمینان بالا و افزایش در دسترس بودن است

ما اکثر تکنیک‌ها و مدل‌های تحمل خطای موجود را که در سرویس رایانش ابری استفاده می‌شوند، مطالعه و بررسی کردیم. ارائه دهندگان ابر می توانند مناسب ترین تکنیک یا مدلی را انتخاب کنند که نیازهای آنها را برآورده کند. تحمل خطا در رایانش ابری هنوز یک حوزه تحقیقاتی فعال است.

منابع

[1] Kumari, P., Kaur, P. A., 2018. Survey of fault tolerance in cloud computing. Journal of King Saud University Computer and Information Sciences.

[2] Mohammad H. Alshayeji, Mohammad Al-Rousan, Eman Yossef, Hanem Ellethy, 2017. A Study on Fault Tolerance Mechanisms in Cloud Computing, Electrical and Computer Engineering (ECE)

[3] Sarabjit Kumar, Dr. Ajay Shriram Kushwaha, 2019. Future of Fault Tolerance in Cloud Computing. THINK INDIA JOURNAL. ISSN: 0971-1260 Vol-22-Issue

[4] Tomas, T., Gupta, B., Carlos, F., 2000. Supply chain management: Theory and Systems, Academic Press, Tehran, 54-63.

[5] Zeeshan A., Nisha S., Harshpreet S., 2015. Review on Fault Tolerance Techniques in Cloud Computing. International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 18,

[6] Mukwevho, Mukosi Abraham, and Turgay Celik., 2018. Toward a smart cloud: A review of fault tolerance methods in cloud systems. IEEE Transactions on Services Computing

[7] Prasenjit Kumar Patra, Harshpreet Singh, Gurpreet Singh, 2013. Fault Tolerance Techniques and Comparative Volume 64 - No.14 Implementation in Cloud Computing International Journal of Computer Applications (0975 - 8887)

[8] Stefan Brenner, Benjamin Garbers, Rüdiger Kapitsa, 2014. Adaptive and Scalable High Availability for Infrastructure (DAIS), Jun 2014, Berlin, Clouds. 4th International Conference on Distributed Applications and Interoperable Systems Germany. Pp.16-30,

[9] Pan Y, Hu N , 2014. Research on dependability of cloud computing systems. In: International conference on reliability, maintainability and safety (ICRMS). IEEE, Piscataway, pp 435–439

[10] Patidar, S., Rane, D., Jain, P., 2012. A survey paper on cloud computing. In: Proc. 2012 2nd Int. Conf. Adv. Comput. Commun. Technol. ACCT 2012, pp. 394–398.

[11] Savu, L., 2011. Cloud computing: deployment models, delivery models, risks and research challenges. In: 2011 Int. Conf. Comput. Manag. CAMAN 2011.

[12] Bokhari, M.U., Shallal, Q.M., Tamandani, Y.K., 2016. Cloud computing service models: a comparative study. In: IEEE Int. Conf. Comput. Sustain. Glob. Dev. INDIACom, pp. 16–18.

[13] Awatif Ragmania , Amina Elomria, Noreddine Abghoura, Khalid Moussaida, MohammedRidaa, Elarbi Badidi, 2020 . The 10th International Symposium on Frontiers in Ambient and Mobile Systems (FAMS 2020) April 6-9, 2020, Warsaw, , International Journal of Supply and Operations Management, 2, 70-79. Poland

[14] Mridula Dhingra , Neha Gupta, 2017. Comparative analysis of fault tolerance models and their challenges in cloud computing. International Journal of Engineering & Technology, 6 (2) (2017) 36-40

[15] Ameen Alkasem and Hongwei Liu, 2015. A Survey of Fault-tolerance in Cloud Computing: Concepts and Practice. ISSN: 2040-7459; e-ISSN: 2040- Research Journal of Applied Sciences, Engineering and Technology 11(12): 1365-1377, 7467.

[16] Engr Mostafijur Rahman, 2013 A Survey On Fault Tolerance Techniques And Methods In Cloud Computing. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 2 Issue 6, June - 2013

[17] Dr. J Meena Kumari, Shaima'a, Ghamdan, 2016. A Study on Fault Tolerance Solution. International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, ICRET - 2016 Conference Proceedings.