


| | | |
|----------------------------|--|--|
| شناسه: ب/ک/۱ | بسمه تعالی |  |
| شماره: تاریخ: پیوست: | فرم تصویب پیشنهادیه پایان نامه دانشجوی دوره کارشناسی ارشد در شورای گروه و دانشکده (ویرایش بهار ۱۳۹۳) | اداره تحصیلات تکمیلی |

مشخصات دانشجو:

نام و نام خانوادگی: علی مرتضوی

شماره دانشجویی: ۹۲۲۲۷۶۴

دوره: نوبت اول نوبت دوم

رشته / گرایش تحصیلی: مهندسی فنآوری اطلاعات/شبکه های کامپیوتری گروه: مهندسی کامپیوتر دانشکده: مهندسی برق و کامپیوتر

نشانی و تلفن: زاهدان - بولوار شهید مزاری - خیابان گلستان - کوچه گلستان ۶ - میلان ۱۰ - پلاک ۱۸ - تلفن ۰۹۳۸۹۳۸۳۸۱۴

مشخصات پایان نامه:

۱- عنوان:

فارسی: بهبود تحمل خطا با استفاده از استراتژی تکرار داده در ذخیره سازی ابری

انگلیسی: Enhancing fault tolerance using data replication strategy in cloud storage

نوع پایان نامه: کاربردی بنیادی توسعه‌ای اولین نیمسال اخذ واحد پایان نامه: نیمسال اول ۹۳-۹۲ تعداد واحد: ۶ واحد

مشخصات استادان راهنما و مشاور^۱:

| مسئولیت | نام و نام خانوادگی | آخرین مدرک تحصیلی /مرتبه علمی | گروه/دانشکده/ دانشگاه یا موسسه | امضاء |
|-------------------|--------------------|-------------------------------|--------------------------------|-------|
| استاد راهنمای اول | کیارش میزانیان | دکتری / استادیار | کامپیوتر /برق و کامپیوتر /یزد | |
| استاد راهنمای دوم | | | | |
| استاد مشاور اول | سجاد ظریف زاده | دکتری / استادیار | کامپیوتر /برق و کامپیوتر /یزد | |
| استاد مشاور دوم | | | | |

این پیشنهادیه در شورای تحصیلات تکمیلی / شورای گروه به تاریخ مورد بررسی و تصویب قرار گرفت.

نام و امضای مدیر گروه مهدی رضائیان

این پیشنهادیه در شورای تحصیلات تکمیلی دانشکده به تاریخ مورد بررسی و تصویب قرار گرفت و

اطلاعات مربوط به آن در سامانه پژوهشگاه علوم و فنآوری اطلاعات ایران ثبت و تایید شده است.

نام و امضای رئیس / معاون آموزشی دانشکده

اصل پیشنهادیه تایید شده باید به اداره تحصیلات تکمیلی دانشگاه ارسال گردد.

^۱ امضای اساتید راهنما و مشاور الزامی است و امضای "از طرف" پذیرفته نمی شود. در صورتی که هر یک از اساتید یاد شده عضو هیات علمی دانشگاه یزد نباشند، ضمن درج نشانی و شماره تلفن آنان، آخرین حکم کارگزینی آن‌ها ضمیمه گردد.

الف) تعریف موضوع (تعریف مسأله، هدف از اجرا و کاربرد نتایج تحقیق):

با پیشرفت فناوری اطلاعات نیاز به انجام کارهای رایانشی در همه جا و همه زمان به وجود آمده است. همچنین نیاز به این هست که افراد بتوانند کارهای رایانشی سنگین خود را بدون داشتن سخت‌افزارها و نرم‌افزارهای گران، از طریق خدماتی، انجام دهند. رایانش ابری آخرین پاسخ فناوری به این نیازها بوده است [۱]. ظهور رایانش ابری که به عنوان بزرگ‌ترین تغییر در فناوری اطلاعات در نظر گرفته می‌شود، همه را، از افراد تا گروه‌ها و شرکت‌های بزرگ، برای استفاده از آن برانگیخته است. در حال حاضر محبوبیت استفاده از ابر به طور گسترده‌ای پذیرفته شده است به طوری که سازمان‌ها در حال جا به جایی سیستم‌های پردازش اطلاعات سنتی خود به سمت سرویس‌های ابر، برای ذخیره‌سازی حجم بالایی از داده‌ها هستند [۲]. با گسترش مداوم رایانش ابری و همچنین ظهور داده‌های بزرگ، این‌که چگونه داده‌ها را به طور موثر ذخیره و مدیریت کنیم و پیوسته کیفیت خدمات را برای درخواست‌های کاربر مهیا سازیم، تبدیل به یک مسئله مهم شده است. فایل سیستم‌های توزیع‌شده نقش مهمی را در مدیریت پایگاه داده‌های رایانش ابری بازی می‌کنند. در حال حاضر چهارچوب ذخیره‌سازی موفق‌ترین فایل سیستم‌های توزیع‌شده شامل فایل سیستم گوگل^۲، سرویس ذخیره‌سازی ساده آمازون^۳ و فایل سیستم توزیع‌شده هادوپ^۴ است. در میان آن‌ها هادوپ یک فایل سیستم متن باز است که فایل سیستم گوگل را شبیه‌سازی کرده است و مقیاس‌پذیری^۵ بهتری ارائه داده است و برای پردازش داده‌های توزیع‌شده در مقیاس بزرگ مناسب است [۳]. اما متأسفانه این که در سرویس ذخیره‌سازی ابر، تعدادی از گره^۶ ها در حین عملیات به هر دلیلی با شکست مواجه شوند یا خراب شوند، حتی در بزرگ‌ترین و پیشرفته‌ترین ارائه‌دهندگان سرویس ابر، اجتناب ناپذیر است. به عنوان نمونه در مارس سال ۲۰۰۹ میکروسافت آزور^۷ به مدت ۲۲ ساعت خاموش بود، در همان سال در اتفاقی مشابه تعدادی از مشتریان ارتباط خود را با سرویس آمازون EC2 به مدت ۵ ساعت از دست دادند، بنابراین یک چالش امنیتی جدی برای سیستم مدیریت رایانش ابری آشکار گردید [۳].

عدم دسترسی به سرویس‌های سیستم‌های ابر، می‌تواند به دلایل مختلفی باشد که در زیر ذکر شده اند [۲].

(۱) خرابی سخت‌افزار، مانند پردازشگر، دیسک سخت، سوکت‌ها و حافظه.

(۲) خطاهای نرم‌افزاری، که باعث خرابی برنامه‌های کاربردی^۸ می‌شود.

(۳) خطاهای شبکه، که به دلایلی چون سرریز سرور و ازدحام شبکه ایجاد می‌شوند.

وجود این خطاها و خرابی‌ها در محیط ابر اجتناب ناپذیر است و اتفاق می‌افتد. اما علی‌رغم وجود این خطاها باید دو عامل مهم قابلیت اطمینان^۹ و قابلیت دسترسی^{۱۰} توسط ارائه‌دهنده سرویس ابر تضمین شود. از این رو وجود یک سیاست تحمل خطا^{۱۱} ضروری به نظر می‌رسد [۲]. در سیستم‌های متعارف، تحمل خطا به معنی تعمیر سریع دستگاه خراب‌شده و یا جایگزینی آن با یک دستگاه دیگر است. اما در سیستم ابر تحمل خطا به معنی توانایی مقاومت کردن در برابر تغییراتی است که به دلایل خطاهای سخت‌افزار و نرم‌افزار و شبکه ایجاد شده‌اند [۲]. برای تحمل خطا، دو سیاست کلی وجود دارد. سیاست پیشگیرانه^{۱۲} که به معنی اقداماتی است که برای پیشگیری از رخ دادن خطا انجام می‌شود، که شامل دو استراتژی مهاجرت پیشگیرانه^{۱۳} و جوان‌سازی نرم‌افزار^{۱۴} است. و سیاست واکنشی^{۱۵} است، که شامل اقداماتی است که بعد از رخ دادن خطا، برای بی‌اثر کردن تاثیر خطا انجام می‌گیرند که شامل سه استراتژی نقطه بررسی و راه‌اندازی مجدد^{۱۶}، ارسال مجدد وظیفه^{۱۷} و استراتژی تکرار^{۱۸} است [۲].

² Google File System (GFS)

³ Amazon Simple Storage Service (S3)

⁴ Hadoop Distributed File System (HDFS)

⁵ Scalability

⁶ Node

⁷ Microsoft Azure

⁸ Application

⁹ Reliability

¹⁰ Availability

¹¹ Fault Tolerance

¹² Proactive Policy

¹³ Preemptive Migration

¹⁴ Software Rejuvenation

¹⁵ Reactive Policy

¹⁶ Checkpointing/Restart

یکی از مهم‌ترین استراتژی‌هایی که برای تحمل خطا در سرویس‌های ذخیره‌سازی ابر به کار می‌رود استراتژی تکرار است. هدف از این پایان نامه، بهبود دادن روشی با استفاده از استراتژی تکرار داده، برای افزایش تحمل خطای سرویس‌های ذخیره‌سازی ابر، در مقابل خرابی‌های دستگاه‌های ذخیره‌سازی به هر دلیلی است. انتظار این است که معیارهای قابلیت اطمینان، دسترس‌پذیری بالا، کاهش هزینه و کارآمدی در سطح قابل قبولی برآورده شوند.

نتایج تحقیق که می‌تواند ارائه یک مکانیسم تحمل خطای مناسب برای سرویس ذخیره‌سازی ابر باشد، می‌تواند در این سرویس‌ها مورد استفاده قرار گیرد و توانایی تحمل و مقاومت در برابر خطاها و خرابی‌های احتمالی که اجتناب‌ناپذیر هستند، را به سیستم بدهند. که وجود یک مکانیسم تحمل خطای مناسب، نه تنها قابلیت اطمینان و دسترس‌پذیری بالا را تضمین می‌کند، بلکه می‌تواند باعث کاهش هزینه‌ها گردد.

(ب) سابقه تحقیق:

در سال ۲۰۱۳ لینگ وو^{۱۹} و همکارانش یک سیستم تحمل خطای داده پویا برای ذخیره‌سازی ابر^{۲۰} (DDFMCS) پیشنهاد کردند [۴]. پیاده‌سازی DDFMCS، تبدیلی سبک وزن از داده با استفاده از هادوپ بود و نتایج آزمایش‌ها نشان داد که این سیستم می‌تواند در فضای ذخیره‌سازی صرفه‌جویی کند، و به بهبود عملکرد دسترسی به داده‌ها کمک کند. DDFMCS به صورت پویا داده‌های مختلف را برای مکانیسم‌های تحمل خطا تعیین می‌کند. مانند نرخ فرکانس دسترسی به فایل‌های ذخیره‌شده در جدول فرکانس دسترسی به فایل، تعداد تبدیلات فایل‌های تحمل خطا و زمانی که در آن فایل‌ها در سیستم ذخیره می‌شوند.

در سال ۲۰۱۱ ییلی ژانگ^{۲۱} و همکارانش یک چهارچوب تحمل خطای بیزانس برای رایانش ابری^{۲۲} (BFTCloud) پیشنهاد کردند [۵]. که از استراتژی تکرار به عنوان پایه تحمل خطا استفاده شده است. علاوه بر آن BFTCloud گره‌های داوطلب را بر اساس مشخصات کیفیت سرویس^{۲۳} و عملکرد قابلیت اطمینان انتخاب می‌کند. به ادعای نویسندگان مقاله آزمایش‌های بزرگ آن‌ها در انواع محیط‌های مختلف ابر، نشان می‌دهد که BFTCloud استحکام^{۲۴} سیستم را هنگامی که بیش از f تا از کل $3f+1$ تا منبع، دچار خطا یا خرابی به دلایلی چون خراب شدن^{۲۵} و خطای رفتار خودسرانه و دلخواه^{۲۶} و غیره را تضمین می‌کند.

اما در سال ۲۰۱۳ جولیا سانتوس ورونس^{۲۷} و همکارانش ادعا کردند که این امکان وجود دارد که با حفظ همه مشخصات و ویژگی‌های الگوریتم‌های BFT سنتی، تعداد منابع تکرار^{۲۸} به $2f+1$ کاهش یابد [۶]. کاهش تعداد تکرارها با استفاده از یک سیستم قابل اعتماد ساده دست یافتنی است، که این به نوبه خود هزینه زیرساخت ابر را کاهش می‌دهد.

در سال ۲۰۱۱ پیتر گاراقان^{۲۹} و همکارانش نیز با توسعه یک چهارچوب به نام FT-FC در مورد کنترل توان و امکان‌سنجی سیستم تحمل خطا بیزانس و متحد کردن‌شان در چهارچوب بحث کردند [۷].

در سال ۲۰۱۰ یوشنگ تان^{۳۰} و همکارانش یک سیستم تحمل خطای بهتر در مقایسه با سیستم‌های تحمل خطای بیزانس سنتی ارائه دادند [۸]. آن‌ها یک سیستم تحمل نفوذ مجازی^{۳۱} سازگار با متدی ترکیبی از مدل‌های تحمل خطا توسعه دادند. که این مدل تحمل خطا، منابع تکرار را به دو دسته فعال^{۳۲} و غیرفعال^{۳۳} تقسیم می‌کند. نتایج نشان می‌دهند که سیستم اجازه تحمل خطای f تا منبع تکرار، را از بین کل $2f+1$ منبع تکرار را می‌دهد. و تضمین می‌کند که در وضعیت بدون نفوذ^{۳۴} با تعداد $f+1$ منبع تکرار فعال سیستم به خوبی کار می‌کند و بقیه f تا تکرار در وضعیت غیرفعال هستند. در حالی که در سیستم سنتی تحمل خطای بیزانس برای تحمل f تا خطا در تکرارها، نیاز به وجود $3f+1$ تا منبع تکرار بود.

¹⁷ Task Resubmission

¹⁸ Replication

¹⁹ Liying Wu

²⁰ Dynamic Data Fault-Tolerance System for Cloud Storage (DDFMCS)

²¹ Yilei Zhang

²² Byzantine Fault Tolerant framework for Cloud Computing (BFTCloud)

²³ Quality Of Service (QOS)

²⁴ Robustness

²⁵ Crash

²⁶ Arbitrary Behaviour Faults

²⁷ Giuliana Santos Veronese

²⁸ Replica

²⁹ Peter Garraghan

³⁰ Yuesheng Tan

³¹ Virtualization intrusion tolerance system

³² Active Replica

³³ Passive Replica

³⁴ Intrusion-free stage.

ج) کلمات کلیدی:

فارسی: تحمل خطا، تکرار داده، ذخیره‌سازی ابری

انگلیسی: fault tolerance, data replication, cloud storage

د) فرضیات (یا سئوالات پژوهشی):

۱. تعداد منابع تکرار (اضافی) را حداکثر تا چه حدی می‌توان کاهش داد، به گونه‌ای که معیارهای قابلیت اطمینان و دسترس پذیری تضمین شوند؟ و چه راه‌حلی برای آن پیشنهاد می‌شود؟
۲. برای کاهش تعداد منابع تکرار، چه محدودیت‌هایی وجود دارد؟
۳. چه عواملی در کاهش هزینه‌ی یک سیستم تحمل خطا، با استفاده از استراتژی تکرار داده، موثر هستند؟ و چگونه می‌توان با بهبود این عوامل، به یک سیستم تحمل خطای مناسب با صرفه اقتصادی رسید؟

هـ) روش تحقیق (مخصوص دانشکده‌های علوم انسانی، منابع طبیعی و هنر و معماری):

و) مراحل اجرای پروژه و زمان بندی:

| شهریور ۹۴ | مرداد ۹۴ | تیر ۹۴ | خرداد ۹۴ | اردیبهشت ۹۴ | فروردین ۹۴ | اسفند ۹۳ | بهمن ۹۳ | دی ۹۳ | آذر ۹۳ | آبان ۹۳ | زمان بندی / مراحل اجرا |
|-----------|----------|--------|----------|-------------|------------|----------|---------|-------|--------|---------|-------------------------------|
| | | | | | | | | | | | جستجوی منابع و مطالعات اولیه |
| | | | | | | | | | | | بررسی مکانیزم های مشابه |
| | | | | | | | | | | | اجرا، شبیه سازی و تحلیل نتایج |
| | | | | | | | | | | | جمع بندی و نتیجه گیری کلی |
| | | | | | | | | | | | تنظیم و نگارش پایان نامه |

ز) فهرست منابع و ماخذ:

- [1] C. Modi, D. Patel, B. Borisaniya, A. Patel and M. Rajarajan, "A Survey on Security Issues and Solutions at Different Layers of Cloud Computing," *Journal of Supercomputing*, vol. 63, no. 2, pp. 561-592, 2012.
- [2] A. Ganesh, M. Sandhya and S. Shankar, "A Study on Fault Tolerance Methods in Cloud Computing," *in proceedings of IEEE International Advance Computing Conference (IACC)*, Gurgaon, pp. 21-22, 2014.

- [3] N. Wang, Y. Yang, Z. Mi, Q. Ji and K. Meng, "A Fault-Tolerant Strategy of Redeploying the Lost Replicas in Cloud," in *proceedings of IEEE 8th International Symposium on Service Oriented System Engineering (SOSE)*, Oxford, pp. 7-11, 2014.
- [4] L. Wu, B. Liu and W. Lin, "A Dynamic Data Fault-Tolerance Mechanism for Cloud Storage," in *proceedings of IEEE 4th International Conference on Emerging Intelligent Data and Web Technologies (EIDWT)*, Xi'an, pp. 95-99, 2013.
- [5] Y. Zhang, Z. Zheng, and M. R. Lyu, "BFTCloud: A Byzantine Fault Tolerance Framework for Voluntary-Resource Cloud Computing," in *proceedings of IEEE International Conference on Cloud Computing (CLOUD)*, Washington, pp. 444-451, 2011.
- [6] G. S. Veronese, M. Correia, A. N. Bessani, L. C. Lung and P. Verissimo, "Efficient Byzantine Fault-Tolerance," *IEEE Transactions on Computers*, vol. 62, no. 1, pp. 16-30, 2013.
- [7] P. Garraghan, P. Townend and J. Xu, "Byzantine Fault-Tolerance in Federated Cloud Computing," in *proceedings of IEEE 6th International Symposium on Service Oriented System Engineering (SOSE)*, Irvine, pp. 280-285, 2011.
- [8] Y. Tan, D. Luo and J. Wang, "Cc-vit: Virtualization Intrusion Tolerance Based on Cloud Computing," in *proceedings of 2nd International Conference Information Engineering and Computer Science (ICIECS)*, Wuhan, pp. 1-6, 2010.

۳- مواد، وسایل و دستگاه‌های مورد نیاز و منبع تأمین:

محل تأمین

نام ماده یا دستگاه

- یک دستگاه کامپیوتر با توان پردازشی بالا و امکانات و نرم افزار های مورد نیاز

۴- تعهد نامه دانشجوی:

اینجانب علی مرتضوی متعهد می‌شوم که با توجه به مفاد این پیشنهادیه به طور تمام وقت، زیر نظر اساتید راهنما و مشاور انجام وظیفه نمایم. ضمناً با اطلاع از اینکه کلیه حقوق مادی و معنوی مترتب بر نتایج حاصل از پایان نامه (اعم از چاپ مقاله، کتاب، ارائه به بخش صنعت و ...) متعلق به دانشگاه یزد خواهد بود از انتشار نتایج حاصل از آن بدون مجوز دانشگاه خودداری نمایم.

تاریخ و امضای دانشجو

* موارد مهم که دانشجویان محترم باید به آن توجه داشته باشند:

- ۱ - دانشجو موظف است با نظر استاد/اساتید راهنما پیشنهادیه پایان‌نامه خود را حداکثر تا ۸ هفته پس از شروع نیمسال سوم به گروه آموزشی تحویل دهد. همچنین دانشجو باید با اعمال تغییرات مورد نظر گروه و دانشکده، پیشنهادیه پایان‌نامه خود را حداکثر تا ۱۰ هفته پس از شروع نیمسال سوم به تصویب شورای گروه و حداکثر تا ۱۴ هفته پس از شروع نیمسال سوم به تصویب شورای دانشکده برساند.
- ۲ - پس از تصویب پیشنهادیه در دانشکده، اطلاعات مربوطه توسط دانشجو باید در سامانه پژوهشگاه علوم و فناوری اطلاعات ایران به آدرس www.irandoc.ac.ir ثبت و توسط مدیر گروه /استادراهنما تایید گردد.
- ۳ - پس از تصویب پیشنهادیه پایان‌نامه، باید نسخه اصلی پایان‌نامه و تاییدیه ثبت پیشنهادیه در پژوهشگاه علوم و فناوری اطلاعات ایران جهت بایگانی در پرونده دانشجو به حوزه تحصیلات تکمیلی دانشگاه ارسال شود.