

Detecção de Envelhecimento de Software em Nuvens Privadas usando o Moving Target Defense

Emerson Felipe da Silva¹, Ivson Galdino Borges², Gustavo Callou¹

¹Departamento de Computação
Universidade Federal Rural de Pernambuco (UFRPE)
Recife – PE – Brazil

²Instituto Federal de Pernambuco (IFPE)
Garanhuns – PE – Brazil

{emerson.felipesilva, gustavo.callou}@ufrpe.br

ivson.borges@garanhuns.ifpe.edu.br

Abstract. *Moving Target Defense (MTD) is a proactive cybersecurity strategy that aims to increase the effort asymmetry between attackers and defenders. To achieve this, it introduces dynamism and uncertainty into the attack surface through continuous system reconfigurations, thereby increasing the complexity and operational cost for malicious agents. In private cloud environments, a common approach to implementing MTD is dynamic virtual machine migration between physical hosts. However, continuous execution of these mechanisms may trigger software aging, characterized by the progressive exhaustion of resources and an increased failure rate in computing systems. In this context, this paper investigates the impacts of operations in cloud computing environments under MTD strategies, with a focus on performance degradation and resource consumption patterns. The results indicate a consistent trend of growth in memory RAM and CPU usage, providing empirical evidence of software aging.*

Resumo. *Moving Target Defense (MTD) é uma estratégia proativa de cibersegurança cujo objetivo é ampliar a assimetria de esforço entre atacantes e defensores. Para isso, introduz dinamismo e incerteza na superfície de ataque por meio de reconfigurações contínuas do sistema, aumentando a complexidade e o custo operacional para agentes maliciosos. Em ambientes de nuvem privada, uma abordagem recorrente para a implementação de MTD é a migração dinâmica de máquinas virtuais entre hospedeiros físicos. No entanto, a execução contínua desses mecanismos pode desencadear o fenômeno conhecido como envelhecimento de software, caracterizado pela exaustão progressiva de recursos e pelo aumento da taxa de falhas em sistemas de computação. Neste contexto, este artigo investiga os impactos de operação em ambientes de computação em nuvem submetidos a estratégias de MTD, com foco na degradação de desempenho e nos padrões de consumo de recursos. Os resultados obtidos indicam uma tendência consistente de crescimento no uso de memória RAM e de CPU, configurando evidências empíricas da ocorrência de envelhecimento de software.*

1. Introdução

A Computação em Nuvem transformou a forma de provisionamento de serviços computacionais ao possibilitar escalabilidade e alta disponibilidade por meio da virtualização de recursos. Entretanto, a execução contínua desses ambientes pode ocasionar o fenômeno de envelhecimento de software, caracterizado pela degradação progressiva do desempenho e pelo aumento da ocorrência de falhas em sistemas de longa duração. Esse problema pode resultar no consumo excessivo de recursos, fragmentação de memória e acúmulo de estados inconsistentes, comprometendo a confiabilidade e a continuidade operacional dos serviços em nuvem [Callou and Vieira 2024]. Essa degradação pode abranger desde plataformas de infraestrutura como o OpenStack [Melo et al. 2017], até sistemas de gerenciamento de bancos de dados [Couto et al. 2023] e ecossistemas contemporâneos de orquestração de contêineres, a exemplo do Docker [Vinícius et al. 2022].

A técnica de *Moving Target Defense* [Matheus Torquato 2020] surge como uma estratégia proativa de segurança e resiliência, baseada na alteração dinâmica de parâmetros do ambiente computacional, como a migração de máquinas virtuais entre hospedeiros. Além de dificultar ações maliciosas, essa abordagem pode impactar o comportamento e o consumo de recursos da infraestrutura.

Diante disso, este artigo investiga os efeitos do envelhecimento de software em um ambiente de computação em nuvem baseado no Apache CloudStack, utilizando a técnica de MTD como mecanismo de cibersegurança. Para isso, foi adotada uma abordagem experimental voltada ao monitoramento do consumo de recursos computacionais e à análise do comportamento do sistema ao longo do tempo. Por fim, o trabalho apresenta os estudos de caso propostos para exemplificar a metodologia adotada para identificar indícios de envelhecimento de software nos resultados obtidos.

O artigo está estruturado da seguinte forma. A Seção 2 aborda os trabalhos relacionados. A Seção 3 apresenta os conceitos fundamentais sobre séries temporais e Detecção de tendências. A Seção 4 descreve a metodologia usada. A Seção 5 discute os estudos de caso e os resultados obtidos. E, por fim, a Seção 6 conclui o trabalho e sugere direções para pesquisas futuras.

2. Trabalhos Relacionados

A literatura menciona que o envelhecimento de software é um fator crítico para a confiabilidade de sistemas computacionais, especialmente em ambientes de execução contínua, como servidores, aplicações em nuvem e sistemas embarcados. As pesquisas na área concentram-se na identificação das causas do fenômeno, na análise de seus impactos e no desenvolvimento de estratégias de mitigação e rejuvenescimento de software.

Os estudos sobre envelhecimento de software são tradicionalmente divididos em duas abordagens: a baseada em medições, que monitora recursos do sistema para detectar sinais de degradação, e a baseada em modelos analíticos, que utiliza técnicas matemáticas, como Cadeias de Markov e Redes de Petri, para estimar métricas de confiabilidade, como o tempo médio até a falha (MTTF). Em ambientes de computação em nuvem, a degradação está frequentemente associada ao gerenciamento de memória e à fragmentação de recursos do sistema operacional.

Mais recentemente, pesquisas passaram a focar em ambientes virtualizados e pla-

taformas de nuvem, como o Apache CloudStack, nos quais a interação entre aplicação, hipervisor e sistema operacional aumenta a complexidade do problema. Nesse contexto, técnicas de rejuvenescimento de software, incluindo reinicialização e migração de máquinas virtuais, têm sido propostas como soluções eficientes para restaurar a integridade do sistema e reduzir os impactos do envelhecimento de software.

O estudo apresentado em [Andrade et al. 2021] investiga o envelhecimento de software em um sistema de classificação de imagens, comparando seu comportamento em ambientes de nuvem e de borda. Por meio de testes de estresse, os autores confirmaram a degradação de desempenho e o esgotamento da memória. Em ambos os cenários, contrariamente às expectativas, o ambiente de nuvem, mesmo com maiores recursos computacionais, apresentou uma degradação de desempenho mais acentuada sob condições de carga de trabalho elevada.

Os autores em [Oliveira et al. 2020] investigam o envelhecimento de software em ecossistemas de virtualização baseados em contêineres. No entanto, não foi o foco do estudo concentrar-se em plataformas de gerenciamento de nuvem fundamentadas em virtualização completa, como, por exemplo, utilizando a nuvem privada com o Apache CloudStack.

A metodologia SWARE, proposta em [Torquato et al. 2017], estabelece um arcabouço rigoroso para a caracterização do envelhecimento e a validação de estratégias de mitigação. O protocolo é estruturado em três fases distintas: (i) fase de estresse, na qual a degradação é acelerada mediante a imposição de uma carga de trabalho intensiva; (ii) fase de espera, dedicada à observação da persistência dos sintomas de degradação na ausência de estímulos externos; e (iii) fase de rejuvenescimento, voltada à restauração do estado limpo do sistema, tipicamente via reinicialização ou migração de processos. Embora os autores tenham validado sua abordagem em nuvens privadas, utilizando a migração de máquinas virtuais como mecanismo de recuperação, o presente trabalho expande essa fronteira ao investigar a intersecção entre o envelhecimento de software e o MTD.

Em [Vinícius et al. 2022], é apresentada uma investigação sobre envelhecimento de software na plataforma Docker. A metodologia empregada integra experimentos de longa duração com ciclos de monitoramento estendidos por aproximadamente 30 dias. Os resultados evidenciaram criticidade no envelhecimento da plataforma em todos os cenários testados, manifestada por um incremento monotônico e consistente no consumo de memória RAM. A relevância desse estudo reside não apenas na análise de sensibilidade ao hardware, mas na introdução de modelos preditivos para a antecipação de falhas. Em convergência com essa abordagem, o presente trabalho monitora métricas de CPU, memória RAM e uso de disco, para a detecção de anomalias de envelhecimento, contudo, a nossa análise insere-se no domínio das estratégias de MTD.

O autores em [Couto et al. 2023] investigaram a degradação de desempenho e o envelhecimento de software em sistemas de gerenciamento de banco de dados PostgreSQL. A metodologia consistiu na submissão do sistema a cargas de trabalho ininterruptas por um período de 48 horas, com o monitoramento contínuo de métricas de CPU e memória RAM. O diferencial desta abordagem reside na aplicação do teste estatístico de Mann-Kendall, um método utilizado para detectar tendências em séries temporais. Os

resultados confirmaram, com significância estatística, uma trajetória ascendente no consumo de recursos em todos os perfis de carga avaliados. Embora o objeto desse estudo seja um sistema de banco de dados, a sua fundamentação estatística é diretamente transponível para a análise de infraestruturas de nuvem. No presente trabalho, é adotada uma abordagem semelhante, empregando métodos estatísticos para validar se as variações observadas configuram tendências reais de envelhecimento ou flutuações temporárias.

Os autores em [Couto et al. 2024] apresentam uma análise comparativa do envelhecimento de software nos sistemas gerenciadores de banco de dados SQL Server e MySQL. Sob perfis de carga heterogêneos, ambos os sistemas manifestaram um incremento progressivo no consumo de memória RAM, acompanhado de uma degradação mensurável nos tempos de resposta. A validação dessas tendências foi realizada por meio do teste estatístico de Mann-Kendall. Embora o escopo de estudo limite-se a SGBDs, a metodologia aplicada é diretamente transponível para a investigação de infraestruturas de nuvem. A adoção do teste de Mann-Kendall em nosso estudo visa conferir o nível de robustez na validação do consumo de recursos no Apache CloudStack. Este método permite distinguir, de forma quantitativa, anomalias causadas pelo envelhecimento de software de flutuações ou ruídos operacionais, assegurando a confiabilidade das conclusões acerca da necessidade de intervenções de rejuvenescimento.

O estudo apresentado em [Melo et al. 2017] investiga os efeitos do envelhecimento de software na plataforma de computação em nuvem OpenStack. A metodologia empregada consistiu em submeter a infraestrutura a um cenário de estresse caracterizado por ciclos iterativos de *start* e *stop* de máquinas virtuais, visando ao monitoramento da exaustão de recursos. As evidências experimentais identificaram um processo crítico de degradação no componente MySQL subsistema essencial do OpenStack, manifestado por um crescimento exponencial no consumo de memória. Embora essa pesquisa utilize protocolos de carga de trabalho semelhantes, o presente trabalho se diferencia em dois pilares fundamentais: O foco sobre o orquestrador Apache CloudStack, e a análise do impacto de estratégias de MTD. Enquanto o estudo anterior foca na carga de trabalho de usuários comuns, este artigo explora como a estratégia de segurança com MTD influencia a taxa de envelhecimento e a estabilidade da nuvem, oferecendo uma visão integrada de segurança e resiliência operacional.

A investigação conduzida em [Avritzer et al. 2025] aborda o envelhecimento de software e o rejuvenescimento em clusters de larga escala compostos por máquinas virtuais heterogêneas. O estudo transcende a mera detecção do fenômeno ao avaliar sistematicamente diversas estratégias de mitigação e ao propor modelos matemáticos para a determinação de políticas ótimas de escalonamento. Uma contribuição fundamental desta pesquisa é a taxonomia das fontes de degradação, classificadas em: envelhecimento intrínseco, decorrente de falhas como vazamentos de memória na aplicação; e envelhecimento ambiental, resultante da contenção de recursos no hospedeiro físico, exacerbada pela competição entre múltiplas instâncias virtualizadas. Os autores demonstram que a eficácia do rejuvenescimento é sensível tanto à granularidade da intervenção quanto ao instante de sua execução. Para equacionar essas variáveis, foi desenvolvido um modelo baseado em Processos de Decisão de Markov, permitindo a derivação de políticas que minimizam custos operacionais e taxas de falha ao selecionar dinamicamente o dimensionamento das instâncias. O presente trabalho preenche esta lacuna ao introduzir o MTD

como fator de influência. Em suma, embora a literatura pregressa tenha explorado o envelhecimento de software sob a ótica de aplicações específicas, como classificadores de imagens [Andrade et al. 2021], ou orquestradores como o OpenStack [Melo et al. 2017], este artigo distingue-se ao propor uma análise do fenômeno de envelhecimento de software na plataforma Apache CloudStack, com foco rigoroso na disponibilidade e na manutenibilidade sistêmica.

A metodologia proposta apresenta uma abordagem distinta ao reproduzir o ciclo de vida operacional de serviços em nuvem, utilizando processos iterativos e realistas de criação e desligamento de máquinas virtuais. Dessa forma, a presente abordagem não apenas possibilita o monitoramento do consumo de recursos críticos, como também estende e adapta protocolos para detectar envelhecimento de software a partir de estratégias de segurança que fazem uso de MTD. Dessa forma, ao investigar como a rotatividade defensiva de instâncias em uma nuvem privada impacta a utilização de recursos do sistema, este trabalho supre uma lacuna crítica sobre a viabilidade de conciliar segurança proativa com estabilidade operacional de longo prazo.

A Tabela 1 apresenta um resumo comparativo dos trabalhos encontrados na literatura sobre envelhecimento de software e estratégias de MTD em diferentes ambientes computacionais. A partir desta tabela, é possível analisar as abordagens empregadas e as principais métricas avaliadas em cada estudo. Observa-se que a maioria dos trabalhos se concentra exclusivamente na análise de desempenho [Melo et al. 2017, Couto et al. 2023, Vinícius et al. 2022, Andrade et al. 2021, Oliveira et al. 2020, Torquato et al. 2017], enquanto apenas alguns investigam tanto o desempenho quanto a confiabilidade [Couto et al. 2024, Avritzer et al. 2025]. Vale destacar que esse trabalho estende trabalhos anteriores [Silva et al. 2026], [Felipe et al. 2025], mas considerando cenários mais complexos e levando em consideração um *baseline* para melhorar a efetiva comparação do impacto do MTD sob as métricas em avaliação. Além disso, alguns estudos são levantamentos puramente teóricos [Cho et al. 2020, Tajmilur Rahman 2022], sem experimentação prática, sendo esse um dos diferenciais do trabalho proposto.

3. Fundamentação Teórica

Esta seção apresenta os conceitos fundamentais sobre séries temporais e detecção de tendências que são adotados para detectar indícios de envelhecimento de software a partir do MTD.

3.1. Séries Temporais

A análise de séries temporais é fundamental para compreender e mitigar o envelhecimento de software, especialmente em ambientes de computação em nuvem, nos quais o comportamento do sistema e o uso de recursos variam continuamente. O monitoramento de métricas de desempenho, como latência, uso de CPU, consumo de memória e tráfego de rede, permite identificar padrões e tendências ao longo do tempo. Alterações progressivas nessas métricas, como degradação de desempenho ou aumento no consumo de recursos, podem indicar precocemente a ocorrência de envelhecimento de software. Além disso, técnicas de previsão baseadas em séries temporais auxiliam na antecipação de falhas e na identificação de componentes mais suscetíveis ao envelhecimento.

Tabela 1. Comparação das características entre os trabalhos relacionados.

Trabalhos	Ambiente	Abordagem/Modelo	Métricas	
			Desempenho	Confiabilidade
[Melo et al. 2017]	Nuvem OpenStack	Instanciação/Finalização de VMs	X	
[Couto et al. 2023]	Banco de Dados PostgreSQL	Baseada em carga de trabalho	X	
[Vinícius et al. 2022]	Plataforma Docker	Metodologia SWARE	X	
[Cho et al. 2020]	Cibersegurança	Revisão sobre MTD		
[Andrade et al. 2021]	Computação em Nuvem/Borda	Testes de Estresse	X	
[Oliveira et al. 2020]	Virtualização de Contêineres	Ciclos de VMs (inicialização/desligamento)	X	
[Torquato et al. 2017]	Nuvem Privada	Metodologia SWARE	X	
[Couto et al. 2024]	SQL Server/MySQL	Baseada em carga de trabalho	X	X
[Tajmilur Rahman 2022]	Geral (Revisão)	Revisão da Literatura		
[Avritzer et al. 2025]	Clusters Heterogêneos de VMs	Modelos Matemáticos	X	X
Nosso trabalho	Apache CloudStack	Estratégias de Segurança e os Impactos da MTD na Computação em Nuvem	X	X

3.2. Detecção de Tendência

Para identificar evidências de envelhecimento de software, utilizou-se o teste de Mann-Kendall [Mann 1945], um método não paramétrico para detectar tendências monotônicas em séries temporais. O teste avalia a ausência ou presença de tendência, sendo esta considerada significativa quando $p < 0,05$; a estatística S indica sua direção (positiva para crescente e negativa para decrescente). Para quantificar a magnitude das tendências, empregou-se o estimador de inclinação de Sen, baseado na mediana das inclinações entre pares de pontos, o que garante robustez a outliers. Após a detecção de tendências significativas, o método foi aplicado com 95% de confiança para estimar a taxa de variação das métricas, permitindo comparar o grau de envelhecimento entre os ambientes analisados. A implementação foi realizada com o pacote Kendall na linguagem R.

4. Metodologia

Esta seção apresenta a metodologia adotada para avaliar o impacto da política de segurança através do MTD no envelhecimento de software em computação em nuvem. O estudo desenvolvido nesta pesquisa foi estruturado em etapas sequenciais para garantir a reprodutibilidade e a clareza metodológica, conforme ilustrado na Figura 1.

O processo inicia-se com o Alinhamento Conceitual, fase na qual foram definidos os objetivos do experimento, os métodos para alcançá-los e a realização de revisões bibliográficas para fundamentar a pesquisa sobre o envelhecimento de software e técnicas de MTD. A metodologia prática subdivide-se nas seguintes etapas essenciais, conforme mostrado a seguir.

A Configuração do ambiente envolve a implantação prática de um cenário experimental. Esta etapa consiste na instalação e configuração da plataforma de nuvem e na criação e parametrização das máquinas virtuais. A infraestrutura física utiliza dois servidores, com as configurações dos hardwares do ambiente experimental adotado descritas

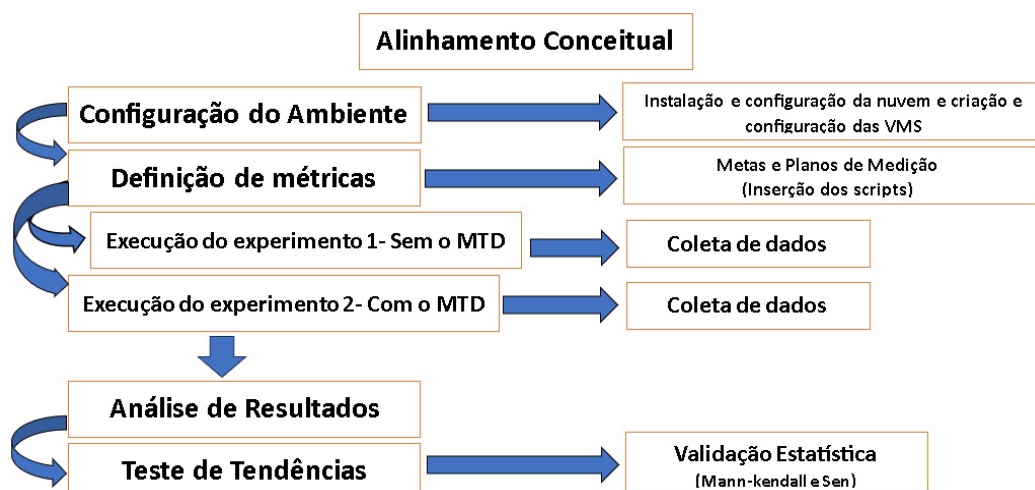


Figura 1. Metodologia

na Tabela 2, e interligados a um switch, com um roteador realizando o isolamento da rede, conforme detalhado na Figura 2.

A definição de métricas para avaliar o comportamento do sistema e os efeitos do envelhecimento de software, estabeleceu um plano de medição com metas claras. Para isso, foram inseridos os *scripts* automatizados responsáveis por simular condições de uso intensivo e monitorar os recursos cruciais de hardware e software.

Para a execução dos experimentos, foi necessária a divisão em dois cenários, de modo a possibilitar a observação e a quantificação da degradação de desempenho ao longo do tempo. O Experimento 1 consiste em um ambiente sem a presença de mecanismos de MTD, sendo utilizado como linha de base para comparação. Já o Experimento 2 é executado em um ambiente sob a influência da estratégia de MTD. Ambos os cenários operam de forma controlada, visando à realização da coleta de dados.

A análise dos resultados e os testes de tendência foram realizados após a etapa de coleta, quando os dados passaram por uma avaliação detalhada. Para garantir a validação estatística das tendências de degradação e do consumo de recursos, foram aplicados os testes estatísticos não paramétricos de Mann-Kendall e o estimador de inclinação de Sen.

Durante os dois experimentos foram realizadas coletas de dados referentes ao consumo de CPU, a utilização da memória RAM e o uso de disco. No experimento 1, as métricas foram obtidas exclusivamente do host 1, enquanto no experimento 2 as medições contemplaram os hosts 1 e 2. As coletas foram executadas por meio de um script em Python¹, de maneira periódica nos servidores. O processo de coleta foi configurado para ocorrer em intervalos de 10 segundos, de forma contínua durante toda a duração dos experimentos. Ao término da execução, os dados obtidos foram compilados e armazenados em arquivos no formato *.CSV*.

A etapa final do estudo compreendeu a análise dos dados coletados. Após o encerramento do período experimental, os arquivos *.CSV* foram submetidos a uma análise detalhada. Para cada intervalo de 60 minutos ao longo das 48 horas de execução, foram

¹<https://github.com/emersonfelipe0813/cloudstack-experimentos/blob/main/coletarmetricas.py>

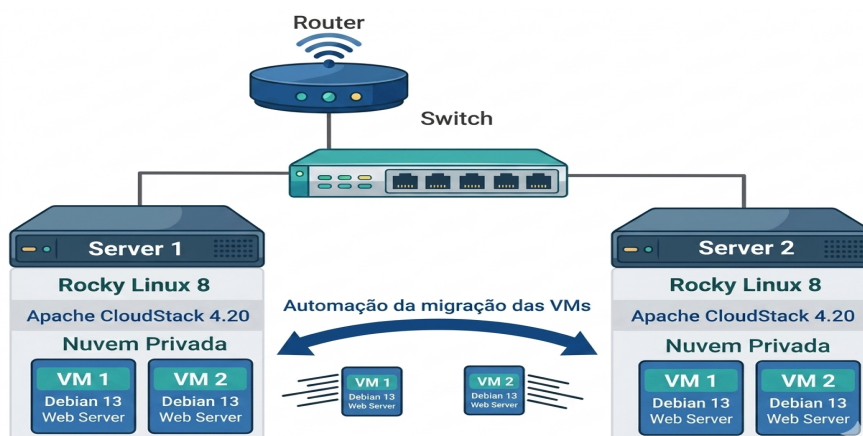


Figura 2. Ambiente de testes

calculadas as médias de consumo de CPU, utilização de memória RAM e uso do disco. A partir desses resultados, foram elaboradas tabelas e representações gráficas com o objetivo de evidenciar a degradação dos recursos computacionais e o fenômeno de envelhecimento de software na plataforma Apache CloudStack. Adicionalmente, visando validar estatisticamente as tendências observadas, foram aplicados os testes de Mann-Kendall e Sen's Slope, os quais permitiram identificar e quantificar tendências monotônicas no comportamento do consumo de recursos ao longo do tempo.

Tabela 2. Configuração dos Servidores

Componentes	Descrição
CPU	Intel® CORE I3®, 3.90 GHz
RAM	8 GB UDIMM RAM
Disco	250 GB SSD
Interface de rede	Gigabit Ethernet
Sistema operacional	Rocky Linux 8

5. Estudos de Caso

Esta seção apresenta dois estudos de caso que demonstram a aplicabilidade da metodologia proposta para avaliar se a estratégia do MTD ocasiona o efeito colateral do fenômeno de envelhecimento de software em ambientes de computação em nuvem. O primeiro estudo concentra-se na medição dos recursos computacionais avaliados, especificamente o consumo de CPU, memória RAM e uso do disco de armazenamento, com o objetivo de identificar sinais de degradação dos recursos. Já o segundo estudo aborda a validação dos resultados do primeiro estudo de caso, por meio de métodos estatísticos, permitindo uma análise mais robusta e confiável do fenômeno de envelhecimento de software.

5.1. Estudo de Caso 1

Com o propósito de estabelecer um *baseline* de referência para as etapas subsequentes, esse estudo investiga se o ambiente computacional, operando em condições normais de uso durante um período contínuo de 48 horas, apresentaria indícios de envelhecimento

de software. Para isso, utilizando apenas o host 1, as duas máquinas virtuais foram instanciadas na nuvem e submetidas a cargas de trabalho sem e com a aplicação da técnica de MTD, de modo a simular condições reais de utilização do ambiente computacional. Durante esse estudo, foram coletadas métricas referentes ao consumo de CPU, utilização de memória RAM e uso do disco de armazenamento do host 1 por meio de um script em Python². Os resultados obtidos, representados nos gráficos da Figura 3, demonstram indícios de envelhecimento de software na utilização de recursos de memória RAM e utilização de CPU em relação ao ambiente baseline sem a execução do MTD. É importante mencionar que, sem o MTD (linha vermelha), pode-se observar claramente que tanto a carga de uso quanto o período de execução do experimento não foram suficientes para provocar degradação significativa no consumo dos recursos do host 1.

Além disso, foi aplicada a técnica de MTD no experimento. Para isso, foi implementado um processo automatizado em Python³, utilizando a API CS do CloudStack, para a migração contínua das máquinas virtuais entre os hosts 1 (linha azul) e 2 (linha laranja) ao longo de um período de 48 horas. Foram coletadas métricas referentes ao consumo de CPU, utilização de memória RAM e uso do disco de armazenamento dos hosts 1 e 2. A execução sucessiva das migrações das VMs demonstrou que essa estratégia de segurança ocasionou degradação no consumo de CPU e na utilização de memória RAM em ambos os servidores. Em relação ao uso do disco, entretanto, a degradação observada apresentou-se de forma menos significativa e pouco perceptível, como mostra os gráficos na Figura 3.

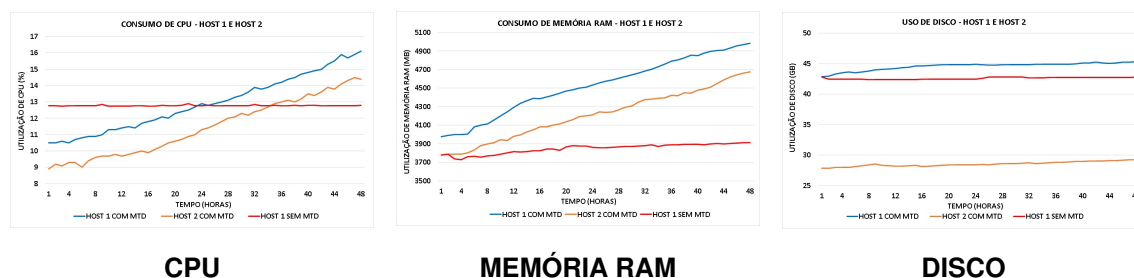


Figura 3. Consumo de recursos com MTD e sem MTD

5.2. Estudo de Caso 2

Para validar os indícios de envelhecimento de software nos resultados obtidos no estudo de caso anterior, foi feita a análise de tendência da série temporal por meio do teste não paramétrico de Mann-Kendall. O teste avalia a hipótese nula de ausência de tendência contra a hipótese alternativa de existência de tendência ao longo do tempo. Para quantificar a magnitude da tendência identificada, foi aplicado o estimador de Sen, que calcula a mediana das inclinações entre todos os pares possíveis de observações da série. As análises foram conduzidas no software R, utilizando o pacote Trend, adotando-se nível de significância de 5%. A análise estatística aplicada aos servidores Host 1 e Host 2, consolidada nas Tabelas 3 e 4, revelou tendências de degradação positivas e estatisticamente significativas em todas as métricas críticas de infraestrutura analisadas ($p < 0,05$). No Host 1, observa-se uma exaustão acentuada de memória RAM, que apresentou a maior

²<https://github.com/emersonfelipe0813/cloudstack-experimentos/blob/main/coletarmetricas.py>

³<https://github.com/emersonfelipe0813/cloudstack-experimentos/blob/main/migracao.py>

taxa de crescimento absoluta do estudo, com uma inclinação de Sen de 20.6060 MB por unidade de tempo ($p < 2,2 \times 10^{-16}$), sugerindo vazamentos de memória. Nesse sentido, o consumo de CPU e o uso de Disco no Host 1 também exibem tendências crescentes estáveis, com inclinações de 0.1200 e 0.0317, respectivamente, conforme ilustrado na Figura 4.

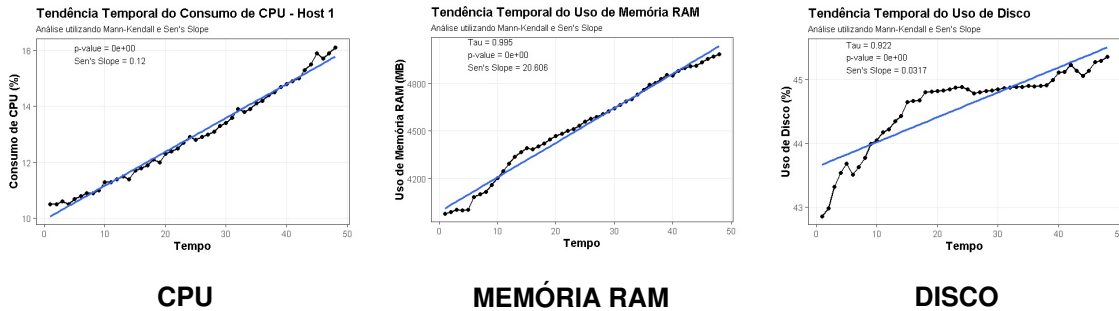


Figura 4. Tendência Temporal - HOST 1

Tabela 3. ANÁLISE DE TENDÊNCIAS - HOST 1

Recurso	Sen's Slope	p-valor	Tendência
Memória RAM	20.6060	$< 2,2 \times 10^{-16}$	Crescente
Uso de CPU	0.1200	$< 2,2 \times 10^{-16}$	Crescente
Uso de Disco	0.0317	$< 2,2 \times 10^{-16}$	Crescente

O Host 2 apresentou um cenário de degradação bastante similar e igualmente preocupante. O consumo de memória RAM manteve uma forte tendência de subida, com inclinação de Sen de 19.0000 MB por unidade de tempo ($p < 2,2 \times 10^{-16}$). Quanto aos outros recursos computacionais do Host 2, o consumo de CPU registrou uma inclinação de 0.1250 ($p < 2,2 \times 10^{-16}$) e o uso de disco obteve uma taxa de crescimento de 0.0275, conforme ilustrado na Figura 5 e detalhado na Tabela 4. As Figuras 4 e 5 apresentam a série temporal juntamente com a linha de tendência estimada pelo método de Sen, evidenciando o crescimento consistente das variáveis ao longo do período analisado. Os resultados confirmam que, embora a técnica de MTD aumente a segurança, ela induz um envelhecimento de software, tornando indispensável a integração de políticas de rejuvenescimento de software para mitigar os efeitos colaterais da própria estratégia de defesa.

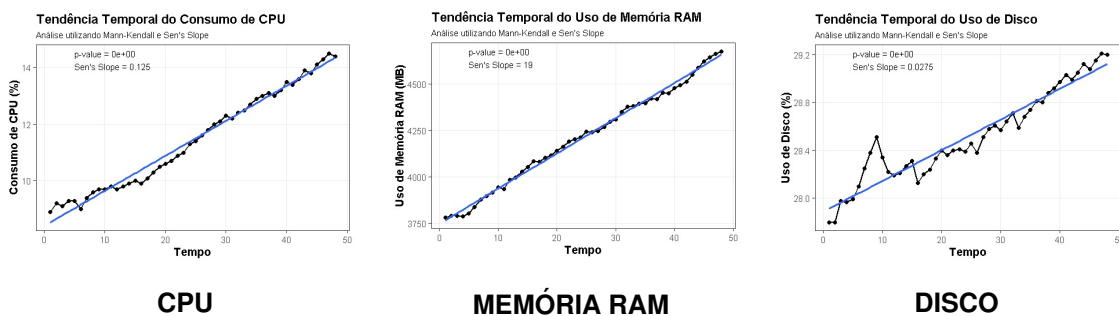


Figura 5. Tendência Temporal - HOST 2

Tabela 4. ANÁLISE DE TENDÊNCIAS - HOST 2

Recurso	Sen's Slope	p-valor	Tendência
Memória RAM	19.0000	$< 2,2 \times 10^{-16}$	Crescente
Uso de CPU	0.1250	$< 2,2 \times 10^{-16}$	Crescente
Uso de Disco	0.0275	$< 2,2 \times 10^{-16}$	Crescente

Dessa forma, uma vez que foi possível identificar indícios consistentes de envelhecimento de software no intervalo observado, optou-se por delimitar o experimento a esse período de amostras. Tal escolha se justifica por sua suficiência em evidenciar o fenômeno investigado, ao mesmo tempo em que permite a otimização dos recursos computacionais e a reprodutibilidade dos testes em um cenário controlado.

6. Conclusão

Este estudo investigou a detecção do envelhecimento de software em ambientes de computação em nuvem que utilizam a técnica de *Moving Target Defense*. Por meio dos estudos de caso realizados, foi possível identificar evidências claras de degradação gradual no desempenho dos hosts, caracterizando o fenômeno de envelhecimento de software decorrente da utilização da estratégia de segurança baseada em MTD. A aplicação de métodos estatísticos, como o teste de Mann-Kendall e o estimador de inclinação de Sen, ambos capazes de detectar tendências sutis de degradação ao longo do tempo, permitiu validar os resultados obtidos e sustentar as conclusões apresentadas neste artigo. Como trabalho futuro, propõe-se a extensão deste estudo por meio da construção de um modelo estocástico baseado em Redes de Petri Estocásticas, com o objetivo de representar o comportamento dinâmico do ambiente de computação em nuvem analisado.

Agradecimentos

Os autores agradecem à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processo de número 309338/2025-5, pelo suporte financeiro para a realização desta pesquisa.

Referências

- Andrade, E., Pietrantuono, R., Machida, F., and Cotroneo, D. (2021). A comparative analysis of software aging in image classifiers on cloud and edge. *IEEE Transactions on Dependable and Secure Computing*, PP:1–1.
- Avritzer, A., Janes, A., Marin, A., Trubiani, C., van Hoorn, A., Camilli, M., Menasché, D., and Bondi, A. (2025). Software aging detection and rejuvenation assessment in heterogeneous virtual networks. *IEEE Transactions on Emerging Topics in Computing*, PP:1–15.
- Callou, G. and Vieira, M. (2024). Availability and performance analysis of cloud services. In *Proceedings of the 13th Latin-American Symposium on Dependable and Secure Computing*, LADC '24, page 262–271, New York, NY, USA. Association for Computing Machinery.

- Cho, J.-H., Sharma, D. P., Alavizadeh, H., Yoon, S., Ben-Asher, N., Moore, T. J., Kim, D. S., Lim, H., and Nelson, F. F. (2020). Toward proactive, adaptive defense: A survey on moving target defense. *IEEE Communications Surveys & Tutorials*, 22(1):709–745.
- Couto, H., Andrade, E., Silva, F. A., and Callou, G. (2023). Analysis of software aging in a database environment. *IEEE Latin America Transactions*, 21:821–828.
- Couto, H., Machida, F., Callou, G., and Andrade, E. (2024). A comparative analysis of software aging in relational database system environments.
- Felipe, E., Borges, I., and Callou, G. (2025). The impact of moving target defense on software aging in cloud computing. In *2025 IEEE 36th International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 01–08.
- Mann, H. B. (1945). Nonparametric tests against trend. *Econometrica: Journal of the econometric society*, pages 245–259.
- Matheus Torquato, M. V. (2020). Moving target defense in cloud computing: A systematic mapping study.
- Melo, C., Araujo, J., Alves, V., and Maciel, P. R. M. (2017). Investigation of software aging effects on the openstack cloud computing platform. *J. Softw.*, 12(2):125–137.
- Oliveira, F., Araujo, J., Matos, R., Lins, L., Rodrigues, A., and Maciel, P. (2020). Experimental evaluation of software aging effects in a container-based virtualization platform. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 414–419.
- Silva, E. F., Borges, I., and Callou, G. (2026). Moving target defense em nuvem: Segurança dinâmica e seus efeitos no envelhecimento de software. In *Anais do XXVII Workshop de Testes e Tolerância a Falhas*, pages 136–149, Porto Alegre, RS, Brasil. SBC.
- Tajmilur Rahman, Joshua C. Nwokeji, T. V. M. (2022). Analysis of current trends in software aging: A literature survey.
- Torquato, M., Maciel, P., Araujo, J., and Umesh, I. M. (2017). An approach to investigate aging symptoms and rejuvenation effectiveness on software systems. In *2017 12th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6.
- Vinícius, L., Rodrigues, L., Torquato, M., and Silva, F. A. (2022). Docker platform aging: a systematic performance evaluation and prediction of resource consumption. *The Journal of Supercomputing*, 78(10):12898–12928.