

Gerenciamento Térmico e Elétrico de um Centro de Dados utilizando Sensoriamento IoT

Antonio Rafael Braga, Felipe Anderson O. Maciel, Rodrigo L. A. Almeida *,
Paulo A. C. Aguiar, Danielo G. Gomes[†], Rossana M. C. Andrade[‡]

¹Grupo de Redes de Computadores, Engenharia de Software e Sistemas (GREat)
Universidade Federal do Ceara (UFC) – Campus do Pici – Bloco 942-A
Av. Mister Hull, s/n – 60455-760 – Fortaleza – CE – Brasil

{antoniobraga, felipemaciel, rodrigoalmeida}@great.ufc.br

{pauloaguiar, danielo, rossana}@great.ufc.br

Abstract. *The advancement of the Internet of Things (IoT) has enabled greater efficiency in monitoring environments, allowing a better management and discovery of the conditions where the application is inserted. This feature is useful for monitoring High Performance Computing (CAD) environments that require tight control of local conditions. The environment management has become essential with the popularization of Cloud Computing, since an inoperative environment is a critical problem. This work presents a thermal and electrical control system for CAD environments using IoT. The system monitors the temperature, humidity and electrical power supply of the data center room.*

Resumo. *O avanço da Internet das Coisas (IoT) possibilitou uma maior eficiência no monitoramento de ambientes, permitindo um melhor gerenciamento e descoberta das condições onde a aplicação está inserida. Essa característica é útil para monitorar ambientes de Computação de Alto Desempenho (CAD), que precisam de um controle rígido das condições do local. A gerência desses ambientes tornou-se algo essencial com a popularização da Computação em Nuvem, pois a inoperância desses ambientes é problema crítico. Neste trabalho, apresentamos o GREat Data Center (GDC), um sistema IoT de gerenciamento térmico e elétrico de ambientes de CAD. O sistema monitora temperatura, umidade e fornecimento de energia elétrica da sala de um data center.*

1. Introdução

A Internet das Coisas (IoT) é um paradigma cujo objetivo é conectar diferentes objetos, sendo eles físicos ou virtuais à Internet de modo que interajam e troquem informações entre si de forma autônoma, a fim de atingir um objetivo comum [García et al. 2014]. Dentre os muitos objetivos que pode-se atingir com a IoT é possível destacar a melhoria em processos de produção de alimentos, na produção e distribuição de energia e no monitoramento de ambientes [Borgia 2014] [da Silva et al. 2016] [Carvalho et al. 2015]. No caso do monitoramento de ambientes, demos destacar o monitoramento de casas, veículos, cidades, plantações e centros de dados.

*Bolsista da Fundação Cearense de Pesquisa e Cultura

[†]Bolsista de Produtividade CNPq em Pesquisa (PQ-2)

[‡]Bolsista de Produtividade CNPq em Desenvolvimento Tecnológico (DT-2)

O uso dos centros de dados em organizações públicas ou privadas se intensificou nos últimos anos, principalmente após a popularização da Computação em Nuvem. Consequentemente, o monitoramento desses ambientes tornou-se algo essencial, haja vista que a inoperância desses ambientes passou a ser visto como um problema crítico. A inoperância pode ocorrer devido a falha de equipamentos, problemas de configuração, aumento na demanda de usuários ou até mesmo devido a falta de fornecimento de energia elétrica [Lu et al. 2016]. Nesse último caso, passa a existir também o risco de prejuízos financeiros devidos a quebra ou desgaste de equipamentos [Chen et al. 2014].

Outro problema que afeta um centro de dados é a falha no sistema de refrigeração dos locais onde os equipamentos estão inseridos. A instabilidade e imprecisão no serviço de refrigeração é prejudicial, pois aumenta a probabilidade de quebra de discos rígidos, de erro na escrita e leitura de dados nos discos rígidos e memória RAM [El-Sayed et al. 2012]. Por isso, a redundância e monitoramento desses serviços é fundamental [Liu et al. 2016]. Contudo, curiosamente, o gerenciamento térmico e de fornecimento de energia elétrica muitas vezes é negligenciado, de tal sorte que muitas vezes não se tem dispositivos para monitoramento remoto desses serviços [El-Sayed et al. 2012].

Diante dessa problemática, este trabalho propõe o GREat Data Center (GDC), um sistema de monitoramento em IoT que busca resolver alguns dos principais problemas enfrentados no monitoramento de ambientes de centros de dados. Para auxiliar no desenvolvimento do GDC foram traçadas as seguintes questões: *(Q1) Como coletar dados de um ambiente de centro de dados de modo eficiente? (Q2) Como disponibilizar essas informações de modo escalável, interoperável e seguro? (Q3) Como tornar as informações coletadas de fácil utilização e intuitivas?* Essas questões tem como objetivo auxiliar no desenvolvimento do GDC de forma a manter o foco em temas significativos para um sistema de monitoramento de um centro de dados.

Este artigo está organizado da seguinte maneira: na Seção 2, são listados os trabalhos relacionados; na Seção 3, é apresentado e discutido o GDC. O detalhamento da arquitetura utilizada é feito na Seção 3.1, onde a questão *Q2* foi tratada através do uso da Computação em Nuvem. Na Seção 3.2 é apresentada a aplicação móvel desenvolvida que trata sobre a questão *Q3*. A coleta de dados no GDC é detalhada na Seção 3.3, onde a questão *Q1* foi tratada com o uso de políticas e o uso de um formato leve para representação dos dados, o JSON; na Seção 4, apresentamos a avaliação realizada; e finalizamos na Seção 5 com nossas conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Em [Ramos and Bianchini 2008], os autores apresentam um sistema chamado C-Oracle que realiza o gerenciamento térmico de um ambiente de centro de dados. O C-Oracle propõe políticas para identificação de emergências térmicas. Em sua principal política, os autores fazem uma verificação sobre os estados de utilização do centro de dados em relação ao processamento e armazenamento. A partir disso, a política proposta realiza balanceamento de carga a fim de diminuir a carga nos servidores mais quentes. Diferente disso, nosso trabalho utiliza também dados do ambiente do centro de dados para realização do gerenciamento térmico, identificando situação de provável pane no sistema de refrigeração ou de fornecimento de energia elétrica e realizando a notificação dos administradores do centro de dados.

Outro trabalho que aborda um tema semelhante é o desenvolvido por [Liu et al. 2016]. Nesse estudo, os autores propõem um sistema inteligente de sensoriamento e controle de temperatura de *data centers* com foco na eficiência energética. O sistema faz o monitoramento interno e externo do ambiente do *data center*. Esse duplo monitoramento permite que o sistema habilite um sistema de ventilação que faz o ar gelado natural circular pelo interior do *data center* quando os parâmetros de temperatura e umidade forem pertinentes. Com isso, é possível desligar algumas máquinas de ar condicionado, realizando um sistema de resfriamento energeticamente eficiente. No entanto, esse tipo de abordagem nem sempre é praticável, já que algumas regiões do planeta possuem altas temperaturas durante todo o ano ou em longos períodos de tempo.

Em [Chen et al. 2014], os autores apresentam o *design* e avaliação do sistema PTEC, um sistema de previsão de controle térmico e energético em centro de dados. O foco do estudo é verificar se o PTEC consegue realizar de maneira eficiente o monitoramento do centro de dados com um baixo custo no uso do sistema de resfriamento e de energia. O sistema coleta os dados tanto dos ventiladores dos servidores, quanto do ar condicionado e verifica se as informações tanto de temperatura quanto de consumo de energia estão dentro de uma margem de segurança para o centro de dados. A partir da coleta das informações e da manipulação das informações pelas camadas da arquitetura do PTEC é feita a gestão da temperatura e do consumo de energia do *data centers*. A avaliação do sistema é realizada por meio de simulação para verificar a eficiência e a escalabilidade do PTEC. A avaliação do PTEC atingiu bons resultados quanto ao desempenho e ao consumo de energia do sistema.

O uso de componentes difundidos no domínio da IoT, como plataformas de prototipagem rápida, em sistemas de monitoramento não é algo inédito. Os principais obstáculos para que essas plataformas se tornem uma força de transformação nos domínios de engenharia, científico e comercial consistem na sua falta de confiabilidade, flexibilidade, escalabilidade, interoperabilidade e na dificuldade de implantação, operação e manutenção a longo prazo [Ferdoush and Li 2014]. O sistema apresentado por [Ferdoush and Li 2014] tenta solucionar esses problemas através de uma rede de sensores sem fio desenvolvida com Arduino, Raspberry Pi e com XBee. Os autores defendem que esse sistema possui um baixo custo de implantação e é altamente escalável tanto em relação aos tipos de sensores, quanto em relação à quantidade de nós sensores.

Os trabalhos listados nesta seção tem como propósito realizar o monitoramento do ambiente no qual um centro de dados está inserido. Porém, uma das diferenças do nosso trabalho em relação aos listados é definição do *status* do ambiente à partir da verificação das condições das grandezas ambientais, tais temperatura e umidade, e do gerador de energia elétrica, com a verificação do funcionamento do mesmo em casos em que o seu uso seja necessário. Os trabalhos mencionados não fazem esse tipo de verificação devido a limitações das soluções proposta e não por limitação técnica. Além disso, outra preocupação deste estudo é a verificação das informações fornecidas pelos sensores. Para isso, é feita uma triangulação dos dados captados do ambiente através de uma verificação das mudanças das grandezas ambientais em pontos distintos do ambiente. Assim, obtemos mais controle sobre as condições do local monitorado e a possibilidade de prever eventos inesperados onde o centro de dados está inserido. Os trabalhos relacionados não fazem a verificação das informações por uma decisão de projeto.

3. GREat Data Center (GDC)

O Great Data Center foi desenvolvido com base nas questões apresentadas na introdução deste estudo. A partir delas foi realizada a elaboração da arquitetura do GDC, assim como a modelagem e a elaboração da interface do GDC. Além disso, foi elaborada a política de leitura de dados do ambiente e o algoritmo de definição do status do ambiente. Todos os componentes do GDC são descritos a seguir.

3.1. Arquitetura do GDC

A arquitetura do GDC, representada na Figura 1, é composta por quatro elementos principais: Sensores, *Gateway*, *Cloud* (Banco de dados e *Web App*) e Aplicação móvel.

Os sensores coletam os dados de temperatura e umidade do ambiente da sala do *data center*, bem como a tensão de dois pontos da rede elétrica. Um dos pontos é contemplado pelo grupo gerador e o outro não. A estratégia de sensoriamento será detalhada na subseção 3.3.

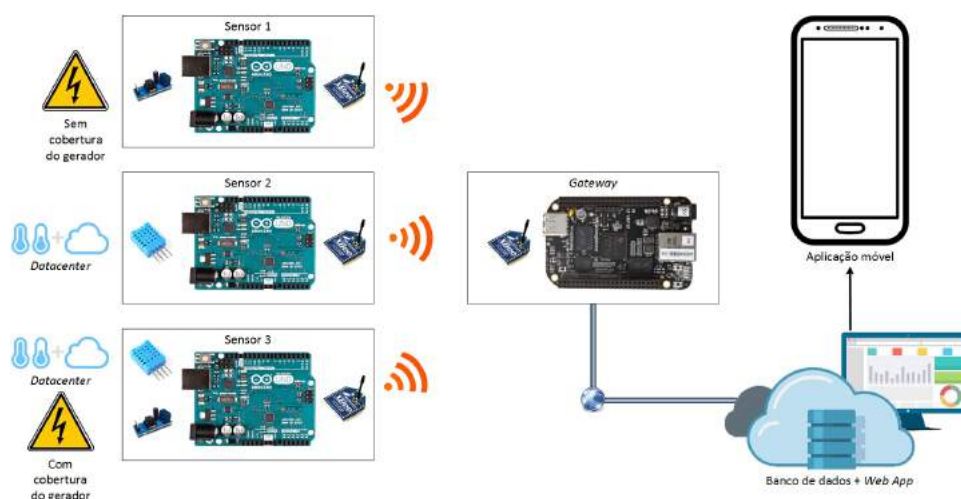


Figura 1. Arquitetura do GREat Data Center.

Os sensores são integrados a kits Arduino, que são equipados com um módulo de rádio XBee.

Os dados obtidos pelos sensores são transmitidos através do protocolo ZigBee para uma placa BeagleBone, que exerce a função de *gateway*. O *gateway* tem como objetivo centralizar os dados coletados. Optou-se por utilizar o ZigBee devido ao seu reduzido consumo energético para transmissão de dados¹. No GDC, o *gateway* tem limitações de processamento, armazenamento e energia.

Nesse cenário, a Computação em Nuvem surge como uma solução para realizar o processamento e armazenamento de forma escalável, além de suprir questões de segurança e controle de acesso. Dessa forma, o *gateway* se comunica com a Nuvem através da pilha de protocolos TCP/IP. O servidor em Nuvem analisa e armazena os dados.

Após o processamento dos dados, as informações geradas são fornecidas aos usuários através de um aplicativo Android.

¹<http://www.zigbee.org/>

3.2. Modelo de Aplicação do GDC

A modelagem do GDC foi feita através do diagrama de casos de usos de acordo com as orientações presentes em [Bezerra 2006]. Conforme os tipos de atores descritos por [Bezerra 2006], o diagrama de casos de usos do GDC possui quatro atores: sensor, *gateway*, administrador e usuário. Cada ator tem uma função relevante em determinada parte do sistema. Por exemplo, o usuário e o administrador tem uma atuação mais relacionada a aplicação móvel, enquanto o sensor está mais relacionado ao subsistema de comunicação. O GDC possui sete casos de uso como demonstrado na Figura 2.

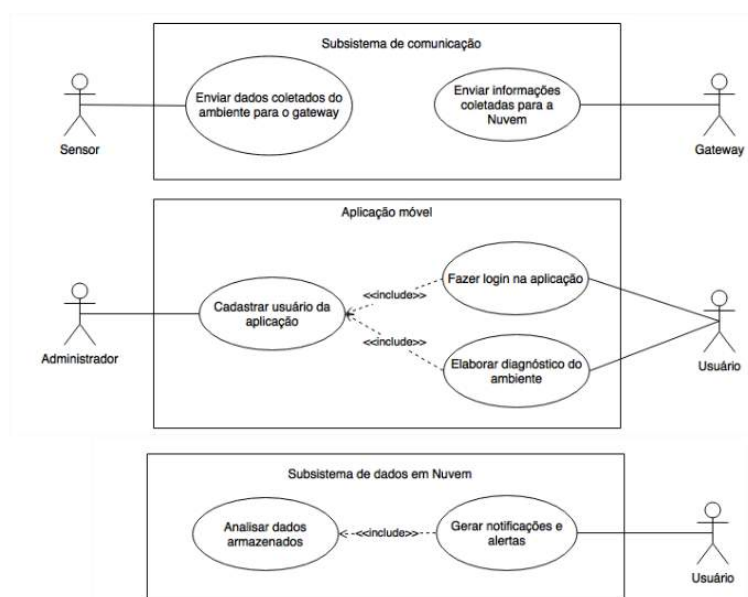


Figura 2. Diagrama de Casos de uso do GREAt Data Center.

Através do diagrama de casos de uso na Figura 2, podemos ter uma visão das funcionalidades que o sistema oferecerá ao usuário. Permitir o acesso a usuários previamente cadastrados, elaborar diagnósticos, notificar o usuário e emitir alertas sobre as condições do ambiente monitorado compõem as principais funcionalidades do GDC descritas no diagrama de casos de uso.

A partir da modelagem do sistema foi possível realizar a elaboração das interfaces do GDC para o aplicativo móvel. A interface do GDC é simples devido a presença de poucos elementos visuais. Além disso, o uso dos ícones, associação de cores a informações apresentadas e legendas propiciam uma compreensão mais eficiente das informações exibidas pelo aplicativo. Essa abordagem é percebida principalmente na tela principal do aplicativo, onde é exibido os *status* do ambiente e os dados de cada sensor. O usuário pode acessar o histórico de medição dos sensores nas últimas 12 e 24 horas. A partir de tal funcionalidade, é possível ao usuário verificar se ocorreu recentemente mudanças bruscas das condições do local monitorado.

Na Figura 3 são apresentadas as principais telas do aplicativo. A figura da esquerda mostra a tela de autenticação do usuário. A figura do meio mostra a gerência dos recursos monitorados, primeiro o status do ambiente, que pode ser seguro, atenção ou alerta. O status é definido através das últimas medições da temperatura, umidade e tensão. Abaixo

do status são apresentadas as últimas medidas das variáveis monitoradas. E na figura da direita é apresentado o histórico das medidas de temperatura.



Figura 3. Principais telas do GREat Data Center.

3.3. Funcionamento do Sistema

A política de coleta de dados do ambiente do GDC possui dois objetivos: (i) reduzir a quantidade de informações enviadas por cada nó sensor a fim de diminuir o consumo energético e (ii) manter a confiabilidade das medições realizadas. A coleta dos dados de interesse é realizada através de sensores conectados a placas de prototipagem rápida Arduino. Os dados de temperatura e de umidade são capturados do ambiente, bem como a presença de tensão na rede elétrica.

A temperatura e a umidade são monitoradas com o intuito de verificar se essas grandezas estão dentro do intervalo ideal determinado pela fabricante dos equipamentos do *data center*. Já a rede elétrica é monitorada para sabermos se o grupo gerador está entrando em funcionamento caso haja uma interrupção no fornecimento de energia. Para isso, monitoramos dois pontos da rede elétrica: um que faz parte da parcela da rede coberta pelo grupo gerador e outro que está fora da cobertura. O monitoramento desses dois pontos está representado pelos Sensores 1 e 3 exibidos na Figura 1. Se houver ausência de tensão nos dois pontos monitorados, saberemos que houve uma interrupção no fornecimento de energia e que o grupo gerador não entrou em funcionamento como deveria.

O sensoriamento dos dados é feito a cada 1 segundo. É importante definir esse intervalo para haver uma economia da energia consumida pelo sensor. Caso contrário, o sensoriamento seria feito na frequência do *clock* do microcontrolador do Arduino, resultando em um alto consumo de energia. Essa é uma das estratégias que contribui para que se cumpra o primeiro objetivo da política.

Após o sensoriamento dos dados, o Arduino envia as informações para o *gateway* através de um módulo de rádio XBee, que utiliza o protocolo de comunicação sem fio ZigBee. O envio das informações é feito quando há alteração nos valores de temperatura ou umidade, ou quando há ausência de tensão na rede elétrica. Essa estratégia de envio e a utilização do protocolo ZigBee, conhecido pela baixa potência de operação, também contribuem para o cumprimento do primeiro objetivo da política. Caso não haja nenhuma alteração nesses parâmetros, as informações são enviadas novamente após um determinado *timeout*, garantindo que a conexão entre o Arduino e o *gateway* esteja sempre ativa.

O segundo objetivo da política é cumprido uma vez que os dados de temperatura e umidade são obtidos por dois sensores distintos, representados pelos Sensores 2 e 3 na Figura 1. Essa redundância na medição converte-se em confiabilidade dos dados. Vale destacar que a precisão do sensor DHT11 foi levada em consideração no tratamento dos dados capturados. Esse sensor possui uma margem de erro de 2°C para temperatura e de 5% para umidade.

As informações coletadas são enviadas para o *Gateway* no formato JSON (*JavaScript Object Notation*). Esse formato foi escolhido por ser leve, de fácil manipulação em diversas linguagens de programação e por permitir a organização das informações de modo estruturado. Cada sensor envia um documento JSON com os dados obtidos.

O *gateway*, ao receber um documento de um dos sensores, apenas armazena o mesmo no banco de dados. Utilizamos um banco de dados baseado em documentos MongoDB². A coleção de cada sensor armazena as últimas medidas alteradas do ambiente e as medidas feitas ao final do *timeout*. Após cada chamada para armazenar um documento, o algoritmo para definição do *status* do ambiente é executado no banco de dados.

Um das principais funcionalidades do sistema é o monitoramento em tempo real das variáveis do ambiente. Essas variáveis monitoradas são processadas a fim de categorizar o *status* do ambiente como SEGURO, ALERTA ou ALARME. O *status* do ambiente é inferido com base nos últimos dados coletados pelos sensores. As medições dos sensores redundantes são comparadas considerando a margem de erro de cada sensor. Para que a medição se torne válida, as medições dos sensores não podem diferir. Caso sejam diferentes, o sistema envia uma mensagem para o administrador informando que há problema na medida dos sensores. Caso as medidas estejam convergindo, então, o algoritmo verifica se o valor lido está dentro da faixa de valores considerados normais (*status* SEGURO) e salva no banco o *status* do ambiente para que a aplicação móvel possa apresentar para o usuário. Caso contrário, salva mensagem informando que o valor obtido pelo sensor está fora do valor aceito como normal para a operação do centro de dados (*status* ALERTA). Se o valor obtido for crítico é emitido um alerta ao usuário. (*status* ALARME)

4. Avaliação

A avaliação desta proposta foi realizada buscando-se analisar o desempenho da aplicação GREat Data Center e verificar os aspectos presentes nas três questões listadas na introdução deste estudo. Para isso observamos as seguintes métricas: uso do processador, memória RAM e rede no *gateway* e tempo para processamento e armazenamento das grandezas monitoradas. Os testes foram executados com três kits Arduínos, um Beaglebone Black com 512 MB de memória RAM e processador com clock de 1 GHz no Centro Nacional de Processamento de Alto Desempenho do Ceará - CENAPAD-CE. Vale destacar que o GDC pode ser implantado em qualquer centro de dados.

Para analisarmos a carga de processamento, o tráfego de dados e a quantidade de memória RAM utilizadas, executamos a aplicação por 8 horas e monitoramos esses parâmetros do sistema através da ferramenta Cacti³. Nesse intervalo de tempo, realizamos

²<https://www.mongodb.com/>

³<http://www.cacti.net/>

8 baterias de teste, cada uma com duração 1 hora. A aplicação foi iniciada por volta das 14:40 horas e executou até 22:40 horas. Foram coletadas 1900 amostras de definição de *status*. Os gráficos gerados pelo Cacti exibem o funcionamento do *gateway* a partir das 13:10 horas para melhor comparação com um período em que a aplicação não estava em execução.

Em relação a carga de processamento, a mesma está representada no gráfico da Figura 4. Percebe-se que a aplicação consome em torno de 20% a mais de processamento normal do *gateway*, o que é aceitável considerando a limitação de recursos do Beagle-Bone. A utilização de memória está representada no gráfico da Figura 5. Consta-se que não houve acréscimo considerável na utilização de memória a partir do início da execução da aplicação.

O tráfego de dados está representado no gráfico da Figura 6. Nota-se que a aplicação aumenta o tráfego de dados em até, no máximo, 15,47 kbits/s, o que é ínfimo para uma rede convencional.

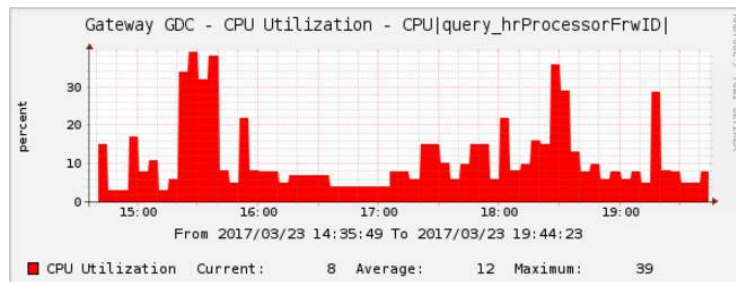


Figura 4. Utilização de CPU durante a execução da aplicação.

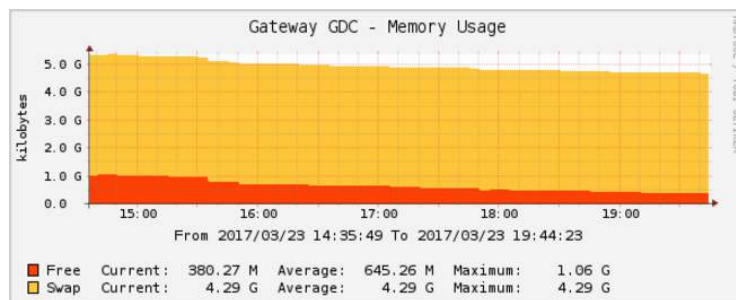


Figura 5. Utilização de memória durante a execução da aplicação.

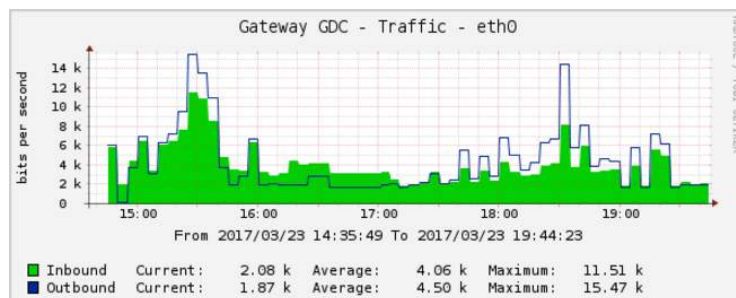
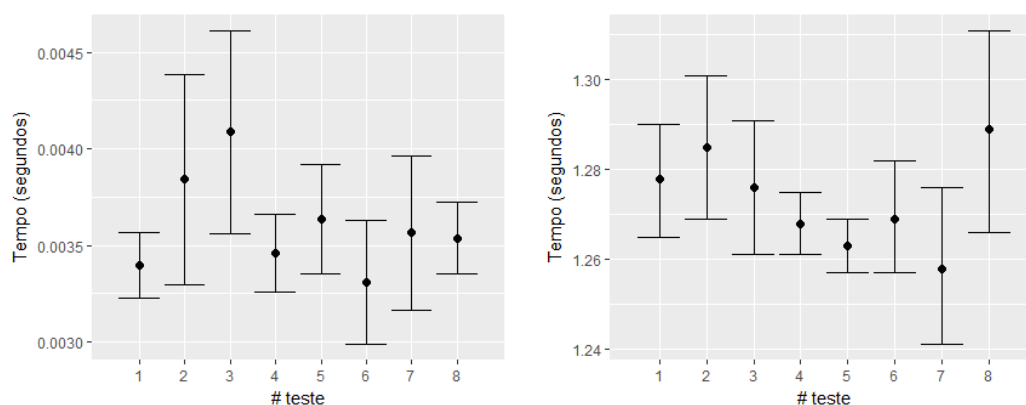


Figura 6. Tráfego de dados durante a execução da aplicação.



(a) Média dos tempos para processamento das informações. (b) Média dos tempos para armazenamento no banco de dados.

Figura 7. Média dos tempos de processamento e armazenamento das grandezas monitoradas.

Os tempos para o processamento e armazenamento das grandezas monitoradas podem ser observados na Figura 7(a) e na Figura 7(b), respectivamente. Foram calculadas as médias dos tempos para cada um dos oito testes realizados, assim como o intervalo de confiança de 95%. Nota-se que os tempos de processamento estão entre 0,0030 e 0,0045 segundos, aproximadamente. Já para o armazenamento, os tempos ficam entre 1,24 e 1,31 segundos, aproximadamente. Esses tempos podem ser considerados satisfatórios, considerando as limitações de recursos do BeagleBone, que processa as informações, e da máquina virtual que hospeda o banco de dados.

A principal contribuição deste trabalho é uma ferramenta de monitoramento para ambientes de centro de dados, utilizando rede de sensores, Computação em Nuvem e um aplicativo móvel. A ideia principal é utilizar os dados gerados pelos sensores para definir um *status* para o ambiente e gerar notificações para o administrador do sistema. A ferramenta monitora temperatura, umidade e tensão elétrica. O uso do formato JSON facilita a inclusão de novas grandezas e torna a coleta de dados mais eficiente (*Q1*). O uso da Computação em Nuvem permite a disponibilização dos dados e informações através de serviços web de modo escalável, interoperável e seguro (*Q2*). A aplicação móvel desenvolvida torna as informações coletadas de fácil utilização e intuitivas (*Q3*).

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou o GREat Data Center, um sistema de monitoramento de centro de dados usando o paradigma de IoT. A partir das questões traçadas, foi possível desenvolver um sistema que apresenta características importantes relacionados a eficiência, segurança e facilidade de uso. A eficiência e segurança do sistema são alcançados a partir das tecnologias utilizadas e da forma de funcionamento do monitoramento que foi desenvolvida. Além disso, a simplicidade e organização das informações na interface podem propiciar uma melhor compreensão do funcionamento do GDC.

Como ponto positivo, destacamos o desempenho do sistema que mostrou-se notável. Como trabalhos futuros, pretende-se realizar a avaliação da proposta quanto aos aspectos como segurança e facilidade de uso. Além disso, outra verificação importante

a ser realizada é quanto à confiabilidade e rapidez de entrega das informações fornecidas pelo sistema ao usuário. Essa verificação é necessária desde o momento em que os dados são coletados pelos sensores até chegar a interface na qual os usuários recebem as informações do sistema. O principal ensinamento que ficou com o desenvolvimento desse trabalho foi que importância do monitoramento do ambiente de centros de dados, que pode ser feita com ferramentas baratas e pode evitar sérios problemas.

Referências

- [Bezerra 2006] Bezerra, E. (2006). *Princípios de Análise e Projeto de Sistemas com UML - 3ª Edição*, volume 3. Elsevier Brasil.
- [Borgia 2014] Borgia, E. (2014). The internet of things vision: Key features, applications and open issues. *Computer Communications*, 54:1 – 31.
- [Carvalho et al. 2015] Carvalho, C. M. A., Rodrigues, C. A. P., Aguilár, P. A. C., de Castro, M. F., Andrade, R. M. C., Boudy, J., and Istrate, D. (2015). Adaptive tracking model in the framework of medical nursing home using infrared sensors. In *2015 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6.
- [Chen et al. 2014] Chen, J., Tan, R., Xing, G., and Wang, X. (2014). Ptec: A system for predictive thermal and energy control in data centers. In *Real-Time Systems Symposium (RTSS), 2014 IEEE*, pages 218–227.
- [da Silva et al. 2016] da Silva, A. R., Júnior, J. C. X., and Silva, I. (2016). Rede de sensores para controle inteligente de ambientes. In *8º SBCUP - Simpósio Brasileiro de Computação Ubíqua e Pervasiva - XXXVI CSBC - Congresso da Sociedade Brasileira de Computação*.
- [El-Sayed et al. 2012] El-Sayed, N., Stefanovici, I. A., Amvrosiadis, G., Hwang, A. A., and Schroeder, B. (2012). Temperature management in data centers: Why some (might) like it hot. *SIGMETRICS Perform. Eval. Rev.*, 40(1):163–174.
- [Ferdoush and Li 2014] Ferdoush, S. and Li, X. (2014). Wireless sensor network system design using raspberry pi and arduino for environmental monitoring applications. *Procedia Computer Science*, 34:103 – 110.
- [García et al. 2014] García, C. G., G-Bustelo, B. C. P., Espada, J. P., and Cueva-Fernandez, G. (2014). Midgar: Generation of heterogeneous objects interconnecting applications. a domain specific language proposal for internet of things scenarios. *Computer Networks*, 64:143 – 158.
- [Liu et al. 2016] Liu, Q., Ma, Y., Alhussein, M., Zhang, Y., and Peng, L. (2016). Green data center with iot sensing and cloud-assisted smart temperature control system. *Computer Networks*, 101:104–112.
- [Lu et al. 2016] Lu, T., Zha, X., and Zhao, X. (2016). Multi-stage monitoring of abnormal situation based on complex event processing. *Procedia Computer Science*, 96:1361 – 1370. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 20th International Conference KES-2016.
- [Ramos and Bianchini 2008] Ramos, L. and Bianchini, R. (2008). C-oracle: Predictive thermal management for data centers. In *2008 IEEE 14th International Symposium on High Performance Computer Architecture*, pages 111–122.