SEGURANÇA NO CONTEXTO DE IOT E FOG COMPUTING

Lukas Derner Grüdtner – 15204126
Aluno da disciplina INE5414 – Redes de Computadores I
do Depto. de Informática e Estatística da Universidade Federal de Santa Catarina

Florianópolis, 30 de Junho de 2017

RESUMO

A Internet das Coisas (IoT – Internet of Things) é uma revolução tecnológica que trouxe dispositivos eletrônicos para o uso em nosso cotidiano, que são equipados com inteligência onipresente. Com essa crescente imersão tecnológica, nos parece cada vez mais inteligente apostar em segurança da informação, pois dados confidenciais transitam pela rede sem a devida proteção, e sua violação pode causar os mais variados tipos de danos.
Com isso em mente, este artigo irá abordar o tema de segurança da informação no contexto de Internet of Things e Fog Computing a nível de survey, explanando sobre o conceito de segurança (confidencialidade, integridade e disponibilidade), principais problemas enfrentados por essas tecnologias, cenários onde se situam e possíveis soluções para um dos principais problemas da atualidade, a segurança dos dados na rede.

1. INTRODUÇÃO

1.1. MOTIVAÇÃO

Com a industrialização e a crescente necessidade de automação e otimização, surgem tecnologias a fim de suprir estas necessidades. Duas delas serão abordadas neste artigo: a Internet das Coisas e a Computação em Névoa. Ambas surgem com objetivos similares, otimizar e melhorar o gerenciamento e processamento de dados, tanto para usuários finais como também para empresas e indústrias, elevando sua produtividade e diminuindo os custos de produção.

Uma rápida análise no Google Trends, uma ferramenta que exibe a frequência que um termo é buscado no Google, exibe a dimensão da popularidade da Internet das Coisas. A segurança não expandiu juntamente com o seu uso, e com uma extensa utilização destes dispositivos, a vulnerabilidade em que nos expomos se torna mais perigosa.

1.2. JUSTIFICATIVA

Com o surgimento de novas tecnologias, surgem com ela também novas vulnerabilidades que não podem ser corrigidas com métodos tradicionais, colocando assim seus usuários em risco. Não existe segurança perfeita e intransponível na tecnologia, pois sempre serão descobertas novas falhas, e novas correções serão aplicadas, e a perpetuação deste ciclo mantém o aprimoramento constante da segurança aplicada nesta tecnologia.

Devido ao grande aumento da automatização em serviços, informações importantes - como dados pessoais, cartões de crédito, ou até informações confidenciais de grandes empresas – percorrem milhares de dispositivos conectados à rede, e muitas vezes estes com um alto grau de vulnerabilidade, tornando-se assim uma necessidade o fator segurança neste contexto.

Por exemplo, o último ciberataque em massa nos Estados Unidos, que afetou pelo menos 85 serviços tais como Netflix, Spotify e redes sociais foi, em parte, culpa de dispositivos IoT. É mais simples instalar um vírus em câmeras conectadas à internet, por exemplo, do que invadir um computador.

1.3. OBJETIVOS

1.3.1. OBJETIVOS ESPECÍFICOS

Este artigo tem como objetivo destacar algumas dessas vulnerabilidades, tanto no contexto de IoT como no de Fog Computing, e ressaltar aspectos importantes e alguns modelos propostos por outros autores para tentar solucionar essas vulnerabilidades. Apresentaremos o desenvolvimento de um projeto proposto por Vishwanath et al (2016) [4], onde algumas técnicas de segurança serão aplicados sobre um determinado conjunto de dados e sua eficácia será posta a prova, mostrando alguns caminhos que podemos seguir para resolver problemas de segurança em Fog Computing. Por fim, serão apresentados tópicos para trabalhos futuros que abrangem este tema, tais como smart grid, a transição para IPv6, a regulamentação do mercado, dentre outros.

1.3.2. OBJETIVOS GERAIS

Como objetivos gerais, pretende-se abranger e explicar sucintamente o conceito sobre a segurança da informação, suas vulnerabilidades e possíveis soluções que possam ser aplicadas a fim de aumentar a proteção dos dados na rede, e também destacar a grande importância que tecnologias como IoT e Fog Computing exercem em nosso cotidiano.

Para o contínuo aprimoramento da segurança, é primordial que as pessoas e organizações entendam a natureza dos ataques aos quais estão expostos. É preciso entender que muitos ataques são resultado da exploração de vulnerabilidades, as quais passam a existir devido a falhas de projeto ou em sua implementação.

1.4. ORGANIZAÇÃO DO ARTIGO

O presente artigo fora organizado da seguinte maneira: na seção 2, são discutidos os principais conceitos básicos sobre a questão da segurança no âmbito de IoT e Fog Computing; na seção 3 são comentados alguns trabalhos correlatos que serviram de base para este artigo; na seção 4 são apresentados alguns dos aspectos relevantes deste tema; na seção 5 são ressaltadas limitações e problemas existentes na segurança da informação; na seção 6 são apresentadas possíveis soluções dadas por autores citados neste artigo; na seção 7 é apresentado o desenvolvimento de uma proposta de projeto feita por um dos autores citados; e, por fim, temos as conclusões obtidas deste artigo e possíveis trabalhos futuros que poderão ser estudados e aprofundados, nas seções 8 e 9, respectivamente.

2. CONCEITOS BÁSICOS

2.1. A INTERNET DAS COISAS

A Internet das Coisas é uma tecnologia muito abrangente que pode ser utilizada praticamente em qualquer área que se possa imaginar. Ele pode estar presente nas casas inteligentes, onde luzes podem ser acesas ao detectar que alguma pessoa entrou no cômodo, ou um ar-condicionado que é ligado automaticamente e, verificando a temperatura externa, ajusta para uma temperatura agradável, ou então medidores inteligentes que monitoram o consumo de energia de cada equipamento da casa; ou nas chamadas cidades inteligentes, onde semáforos podem ter sensores conectados à ambulâncias, que serão ainda conectados a outros semáforos, e, ao perceber uma ambulância se aproximando, poderão simplesmente fechar algumas pistas a fim de deixar o caminho livre para sua passagem.

Em 2009, num artigo publicado através do RFID Journal, Kevin Ashton cita o que é tido por muitos como a definição de IoT: *"...Se tivéssemos computadores que soubessem de tudo o que há para saber sobre coisas, usando dados que foram colhidos, sem qualquer interação humana,*

seríamos capazes de monitorar e mensurar tudo, reduzindo o desperdício, as perdas e o custo. Gostaríamos de saber quando as coisas precisarão de substituição, reparos ou atualização, e se eles estão na vanguarda ou tornaram-se obsoletas.”.

2.2. FOG COMPUTING

A Computação em Névoa (Fog Computing) é uma extensão da Computação em Nuvem (Cloud Computing), pois ela foi concebida com foco nos dispositivos que estão na borda da rede, mais próximas do usuário e assim mais descentralizada, por isso o nome, nevoeiro, uma nuvem que está mais próxima dos usuários.

Com o surgimento da Internet das Coisas, muito tem se planejado sobre como realizar corretamente o processamento de todas as informações geradas pelos dispositivos IoT. E é justamente sobre isso que se trata o conceito de Fog Computing. Seu objetivo é fazer com o que o processamento dos dados gerados ocorra diretamente no equipamento, ou no máximo em algum dispositivo “central” próximo dos dispositivos IoT, sem a necessidade do envio destes dados para a nuvem, evitando um possível sobrecarregamento da mesma.

Fog Computing é um termo que foi criado pela própria Cisco, que viu a necessidade do mercado quando começou a investir em pesquisas relacionadas à Internet das Coisas.

Segundo Stojmenovic et al (2014) [1], a computação em nuvem trouxe muitas oportunidades para as empresas, fornecendo aos seus clientes uma gama de serviços de computação. O atual modelo de computação em nuvem “pague-por-uso” tornou-se uma alternativa eficiente para possuir e gerenciar centros de dados privados para clientes que utilizam aplicativos Web e processamento em lote. A computação em nuvem libera as empresas e seus usuários finais da especificação de muitos detalhes, como armazenamento, limitação de computação e custo de comunicação de rede. No entanto, isto torna-se um problema para as aplicações sensíveis à latência. Quando os dispositivos da IoT estão se envolvendo cada vez mais na vida das pessoas, o paradigma atual da computação em nuvem dificilmente pode satisfazer suas necessidades de suporte à mobilidade e baixa latência. Devido a estes problemas, surge então a computação em névoa. Como ela é implementada na borda da rede, ela fornece baixa latência para aplicações de tempo real.

2.3. SEGURANÇA NO ÂMBITO DE IOT E FOG COMPUTING

A segurança da informação diz respeito à proteção de determinados dados, objetivando a preservação de seus valores para uma empresa ou um indivíduo. Podemos entender como informação todo o conteúdo com capacidade de armazenamento ou transferência, que serve a determinado propósito e que tem alguma utilidade para o ser humano.

Toneladas de informações trafegam todos os dias pela rede, sendo grande parte destas originadas por dados recolhidos por dispositivos de borda, e eles precisam estar protegidos contra ataques. Em redes sem fio, as informações são extremamente vulneráveis e, sem um mecanismo seguro contra invasões, será um prato cheio para os invasores. Com isso surge uma enorme necessidade de segurança das informações.

O acrônimo CIA (Confidentiality, Integrity and Availability) representa os principais atributos da segurança de informação – confidencialidade, integridade e disponibilidade – que orientam a análise, o planejamento e a implementação da segurança para a proteção de determinado grupo de informações. Outros atributos importantes são a irretratabilidade, a autenticidade e a conformidade. Com o rápido crescimento do e-commerce (comércio eletrônico), a privacidade é também uma grande preocupação. A *confidencialidade* define que o acesso das informações são limitadas àquelas autorizadas pelo proprietário da informação; a *integridade* garante que a informação manipulada mantenha todas as propriedades originais estabelecidas pelo proprietário da informação; a *disponibilidade* garante que a informação esteja sempre disponível para usuários autorizados por seu proprietário; a *autenticidade* refere-se a garantia de que a informação é original e não foi manipulada e distorcida ao longo de um processo; a *irretratabilidade* é uma propriedade

que garante a impossibilidade de negar a autoria em relação a uma transação anteriormente feita; e a *conformidade* garante que o sistema deve seguir as leis e regulamentos associados a este tipo de processo.

"De acordo com um estudo da Hewlett Packard, cerca de 70% dos dipositivos da IoT são vulneráveis a ataques. […] O relatório da Hewlett Packard também destacou que informações como cartões de crédito, números de segurança social e outros dados sensíveis percorrem a rede sem a devida segurança." (Gaona-Garcia et al, 2017 [2]).

3. TRABALHOS CORRELATOS

3.1. Security challenges of the Internet of Things (2016)

Weber et al (2016) [3] apresentam em seu artigo os desafios de segurança na área de IoT (Internet of Things), com ênfase em desafios regulamentares que irão surgir nos próximos anos nesta área. Ele também fornece uma definição de IoT e conceitos relacionados, uma visão geral de dois modelos de referência para a arquitetura IoT e os protocolos mais importantes para a Internet of Things. Também é apresentada uma visão geral da atual situação da IoT na Europa, incluindo a situação na Croácia e nos EUA, com os desafios e possíveis problemas para a realização da IoT.

Os autores discutem sobre as várias aplicações da IoT, como o *uso na área econômica*, onde os sensores são usados em fábricas para o trabalho da automação de processos; o *uso pessoal*, no qual as informações coletadas pelos sensores são usados apenas pela pessoa que possui a rede, onde ele cita um exemplo: o e-health care (onde sensores são colocados no corpo do usuário, a fim de monitorar e coletar dados médicos, como a pressão cardíaca, por exemplo, o que permite que pacientes possam ser monitorados em casa, tornando assim mais barato o custo do paciente, já que ele não precisa permanecer no hospital para observação); o *uso para serviços*, onde dados são coletados para buscar a otimização de processos, onde podemos citar um exemplo desse uso: os medidores inteligentes, que permitem aos provedores de serviços gerenciar seus recursos de modo a obter uma maior otimização de seus custos e lucros; e por fim o *uso em dispositivos móveis*.

Dentre os principais desafios de segurança para a área de IoT, os autores citam os seguintes: privacidade de segurança e confiabilidade, padronização, a grande limitação da capacidade de rede e a gestão de grandes quantidades de dados para garantir informações. Além dos desafios citados acima, existem também outros, como: a regulamentação do mercado, a concepção de uma arquitetura mais eficiente para a ligação em rede dos sensores e o armazenamento dos dados recolhidos, o desenvolvimento de mecanismos para o processamento do fluxo de dados recolhidos em redes sensoriais, transição para IPv6, fontes de alimentação de dispositivos/sensores e a redução do custo dos componentes IoT.

3.2. Security in Fog Computing through Encryption (2016)

Em Vishwanath et al (2016) [4], é explicitado o problema da segurança na Computação em Névoa (Fog Computing) através da criptografia usando o algoritmo AES. O autor utiliza no experimento o Android Mobile, e é aplicado a técnica de criptografia analisada sobre três conjuntos de dados de diferentes tipos. São avaliados o desempenho da criptografia, o tempo de resposta e a utilização da memória sobre o tamanho do arquivo. Assim, é avaliado a adequação do algoritmo AES em Fog Computing.

Eles explicam que, no sistema atual, o Decoy é considerado como um modelo de segurança, onde o usuário do sistema tem que, primeiramente, realizar um cadastro e, em seguida, fazer o login. Uma vez que ele tenha se conectado, ele precisa responder a uma pergunta de segurança que foi dado durante a criação da conta. Este é outro método para "fisgar" os atacantes, enganando-os ao mostrar arquivos com nomes e informações falsos, onde apenas o usuário sabe sobre sua falsidade, e o atacante não saberá a diferença entre os dados falsos e os originais. Uma vez que ele clica no arquivo e tenta baixá-lo, o sistema será notificado sobre o invasor, e portanto as

informações não serão roubadas. Os autores explicam que este método de segurança não é adequado pois existe o risco de, ao responder a pergunta de segurança, o invasor (ou qualquer pessoa que conheça o usuário muito bem) possa também responder à pergunta e roubar os dados. Eles sugerem, então, a utilização do algoritmo AES (Advanced Encryption Standard), onde os dados serão criptografados de modo que, mesmo que o invasor acesse os dados da arquitetura atual do sistema chamariz, será improvável que ele consiga lê-los. Em outras palavras, o principal objetivo desta pesquisa é fornecer segurança na segunda camada do sistema cloud-fog usando a técnica de criptografia AES.

3.3. The Fog Computing Paradigm: Scenarios and Security Issues (2014)

Stojmenovic et al (2014) [1] abordam em seu artigo cenários da computação em névoa (Fog Computing) e problemas de segurança no contexto de redes inteligentes e comunicação entre máquinas. Eles explicam que "As principais questões de segurança são a autenticação em diferentes níveis de gateways, bem como (no caso de redes inteligentes) nos medidores inteligentes instalados na casa do consumidor. Cada medidor inteligente tem um endereço IP. Um usuário mal-intencionado pode adulterar seu próprio medidor inteligente, relatar falsas leituras ou falsificar endereços IP.".

Eles sugerem a aplicação de técnicas de detecção de intrusos com base em assinatura, onde os padrões de comportamento são observados e verificados em um banco de dados. A invasão também pode ser detectada usando métodos baseados em anomalias, onde o comportamento observado é comparado com o comportamento esperado, a fim de verificar se houve um desvio.
Um exemplo dado é o ataque do homem do meio. Este ataque pode ser muito silencioso neste contexto, pois ele consumirá apenas uma pequena parcela de recursos dos dispositivos de névoa, como a utilização desnecessária da CPU e o consumo de memória. Por isso, é realmente difícil perceber este tipo de ataque. Um outro problema relatado pelos autores é a questão da privacidade. Um futuro trabalho dos autores será expandir o paradigma de computação em névoa no contexto das redes inteligentes.

3.4. Analysis of Security Mechanisms Based on Clusters IoT Enviroments (2017)

Gaona-García et al (2017) [2] apresentam uma visão geral dos desafios apresentados nos níveis de segurança da IoT. É proposto uma infraestrutura de segurança para neutralizar as vulnerabilidades no IoT utilizando mecanismos como o PKI (Public Key Infrastructure), que possibilita a autenticação da identidade baseada em uma chave pública combinada, solucionando a quantidade excessiva de autenticações. Também é proposto um modelo de 3 camadas, sendo elas: sensor, transporte e aplicação, permitindo a análise de cada um dos componentes envolvidos no processo.

Os autores criaram uma categorização das questões e tecnologias em relação a IoT, que são as seguintes: comunicação, sensores, atuadores, armazenamento, dispositivos, processamento, localização e rastreamento, e identificação; onde são detalhadas cada uma destas categorias e, a partir desta classificação, critérios são definidos para determinar a relevância do nível de segurança em cada uma das áreas.

Como trabalho futuro, os autores preveem realizar uma caracterização de problemas de comunicação, como ataques DDoS, a integridade dos dispositivos, que devem estar protegidos de malwares (como spyware ou rootkits), e mecanismos aplicáveis na privacidade da Internet das Coisas.

3.5 Considerações

O presente artigo aborda o tema de segurança em dispositivos IoT e Fog à nível de survey, retratando os conceitos de Internet of Things, o que ela é, como ela vem sendo utilizada e quais são as suas previsões para o futuro, assim como também para os dispositivos de Fog Computing, e a relação destas duas tecnologias com a segurança da informação. Aborda-se os aspectos relevantes da segurança neste contexto, seus conceitos e principais propriedades, a sua importância, seus efeitos em nossas vidas e o seu estado atual na tecnologia moderna. Problemas existentes nesta área são descritos e observados, tais como as principais vulnerabilidades e possíveis ataques, contextualizando-se com os problemas tratados individualmente nos artigos correlatos, que os descrevem e apontam possíveis soluções que podem ser aplicadas para corrigi-los. Por fim, mostramos uma das propostas de projeto realizadas por Vishwanath et al (2016) [4] que, através de alguns testes sobre um conjunto de dados, analisa o desempenho da criptografia em dispositivos Fog, e também possíveis caminhos para trabalhos que poderão ser realizados futuramente nesta área, indicados pelos autores destes artigos.

4. ASPECTOS RELEVANTES

A segurança por si só já é um aspecto extremamente importante e necessária em nossas vidas cotidianas, tanto a nossa própria segurança física quanto a segurança no mundo virtual. Em muito pouco tempo, a tecnologia dominou a maior parte das atividades que costumávamos realizar pessoalmente, tais como sair de casa para conversar com outras pessoas, sair para ir em bancos pagar contas e boletos, sair para ir ao cinema para assistir filmes, dentre tantas outras atividades que sofreram com mudanças vindas com a tecnologia. Com o advento dos computadores e das redes, tornou-se possível conversar com outras pessoas diretamente pelo computador, podemos até ouvir suas vozes e observar suas expressões direto de um monitor, podemos também pagar contas e assistir filmes sem precisar nos deslocar até aos bancos ou cinemas. A tecnologia tem nos permitido uma maior agilidade e flexibilidade em diversas atividades rotineiras que costumavam tirar-nos algum tempo, nos permitiu uma maior aproximação e conectividade com qualquer pessoa do planeta, bastando apenas um dispositivo com acesso à internet e a conexão será instantânea. Mas com todas essas vantagens e benefícios surgem também novos problemas para a sociedade em geral. Diferentemente de antigamente, onde roubos eram, em sua maioria, praticados físicamente, agora podemos também ser vítimas de roubos virtuais. Ao fazermos compras em lojas virtuais, por exemplo, enviamos dados pessoais, senhas e números de cartão de crédito para realizar a transação, e toda essa informação confidencial trafega pela rede pública, onde qualquer pessoa com um pouco de conhecimento técnico poderá capturar essa informação para interesses próprios. Existem protocolos de segurança, tais como criptografia, envolvidos neste caso, mas a tecnologia está em constante aprimoramento, e o modo como ela é utilizada depende de quem a usa. Portanto surgem também novos métodos para burlar sistemas de seguranças utilizados atualmente. Com novas tecnologias, surgem novas vulnerabilidades, as quais devem ser estudadas e corrigidas o mais cedo possível, antes que possíveis vítimas possam ser alcançadas.

A Internet of Things é uma área em que a pesquisa está em pleno andamento. Após a pesquisa básica sobre tecnologias utilizadas na área de IoT, é necessário introduzir padrões para arquitetura, plataformas e comunicação entre componentes individuais. De acordo com Weber et al (2016) [3], o desenvolvimento de padrões e plataformas para IoT é uma base para o desenvolvimento de serviços avançados. Há muitas direções possíveis para a pesquisa porque a fase madura de desenvolvimento de TI traz novos desafios relacionados à regulamentação do mercado, pagamento, segurança e melhoria de desempenhos e eficiência de trabalho de plataformas. Até agora, a investigação centrou-se no estabelecimento de plataformas funcionais que permitirão a prestação de um número crescente de serviços, mas, com o aumento do número de usuários, será necessário prestar apoio à qualidade de serviço. Os problemas abertos na área de processamento de dados incluem a padronização da linguagem de consulta, a definição de parâmetros-chave de

avaliação (e comparação) para vários componentes e o ajuste de algoritmos para processamento de fluxo de dados em computação em nuvem.

A eficiência energética também representará um desafio significativo no desenvolvimento de protocolos e dispositivos de comunicação. As fontes de energia e a operação autônoma serão de grande importância, em particular para os módulos sensoriais que serão, na maior parte, alimentados por bateria.

A aplicação econômica dos dispositivos IoT no ambiente de trabalho refere-se à "rede de coisas". Os dados coletados por essa rede são usados apenas pelo usuário da rede e podem ser publicados seletivamente. Os sensores sempre foram parte integrante das fábricas em relação à segurança, automação e outros processos necessários para o funcionamento contínuo. Os sensores serão substituídos no futuro por sistemas sem fio, o que garantirá mudanças flexíveis de configurações, ou seja, mudanças mais simples sempre que necessário. O novo sistema é apenas uma sub-rede IoT que é restrita a uma fábrica.

Os dados coletados da rede são mais utilizados para a otimização de processos. Exemplos desse uso incluem medidores inteligentes. Os dados coletados por meio de contadores inteligentes permitem que os provedores de serviços públicos gerenciem seus recursos visando obter uma maior otimização de custos e lucros. Tais sistemas consistem em redes muito caras usadas para a supervisão de infraestrutura chave e gerenciamento eficiente de recursos. A rede móvel, a rede Wi-Fi ou a comunicação por satélite podem ser utilizadas como uma rede-chave. A aplicação concreta das realizações nesta área melhorará significativamente os sistemas de monitoração existentes, que serão mais eficazes no controle de determinados objetivos, notando atividade suspeita e monitoração do acesso não autorizado. Assegurar a qualidade da água ou monitorar as terras agrícolas são apenas algumas das áreas em que a IoT já teve muitos benefícios e é certo que os benefícios da utilização da tecnologia IoT serão ainda maiores no futuro.

As redes de sensores sem fio tradicionais ficam aquém de aplicações que vão além de detecção e rastreamento, mas exigem que os atuadores exerçam ações físicas como abertura, fechamento ou até mesmo transporte de sensores. Neste cenário, atuadores servindo como dispositivos Fog podem controlar o próprio processo de medição, a estabilidade e os comportamentos oscilatórios criando um sistema em malha fechada. Por exemplo, no cenário de trens automantidos, a monitoração de sensores no rolamento das rodas de um trem pode detectar níveis de calor, permitindo que as aplicações enviem um alerta automático ao operador do trem para parar o trem na próxima estação para manutenção de emergência e evitar o descarrilamento em potencial.

5. PROBLEMAS EXISTENTES

Diversas pesquisas mostram que o investimento em segurança feito pelas empresas não chega nem perto do suficiente quando comparado com todos os riscos que estão envolvidos. Devido ao mercado "premiar" aqueles produtos que chegam antecipadamente às prateleiras, as empresas acabam investindo mais em publicidade do que no produto em si, causando, assim, um maior interesse das pessoas por aquele produto. Mas algum tempo após o lançamento, vão sendo descobertos bugs e falhas de segurança por usuários, tanto bem-intencionados quanto mal-intencionados, as quais podem ser utilizadas para roubo de dados, má utilização do serviço ou alguma outra forma ilícita e ilegal de obter vantagens através das mesmas, que podem prejudicar outros usuários do serviço. As empresas esperam que as falhas sejam identificadas após o lançamento, para que então comecem a corrigi-las, o que as poupa tempo durante o seu desenvolvimento. Antigamente, empresas não investiam muito em qualidade de seus produtos, optavam pela quantidade, mas hoje já se sabe que, para uma boa reputação, a empresa deve ter qualidade em seus produtos. O mesmo deverá ocorrer com a segurança.

Um estudo realizado pela HP (Internet of things research study [16]), que considerou dispositivos IoT tais como televisores, webcams, alarmes residenciais, controles de portas e portões, termostatos domésticos, tomadas de energia remotas, balanças e outros equipamentos, mostrou 250

tipos de vulnerabilidades referentes aos dispositivos. Estima-se que até 2020, mais de 34 bilhões de dispositivos IoT serão instalados. Esse aumento na demanda está pressionado os fabricantes para que eles tenham produtos comercializáveis rapidamente, juntamente com capacidades de acesso em nuvem e aplicativos móveis para ganhar compartilhamento. Esse aumento no número de dispositivos abre as postas para ameaças se segurança que vão desde vulnerabilidades de software até ataques de negação de serviço (DoS) a senhas fracas e vulnerabilidades de scripts entre sites. Dentre os problemas encontrados, destacam-se:

- *Problemas de privacidade:* 90% dos dispositivos testados coletaram pelo menos uma informação pessoal através do próprio produto, pela nuvem ou pela sua aplicação móvel.

- *Autorização insuficiente:* 80% dos dispositivos testados não exigiram senhas de complexidade e comprimento suficientes, com a maioria dos dispositivos permitindo senhas como “1234”.

- *Falta de criptografia de transporte:* 70% dos dispositivos testados não criptografaram as comunicações na internet e na rede local.

- *Proteção de software inadequada:* 60 por cento dos dispositivos não usaram criptografia ao baixar atualizações de software. Alguns downloads podem ser interceptados, extraídos e montados como um sistema de arquivos no Linux, onde o software pode ser visualizado ou modificado.

As questões de segurança e privacidade – apresentadas por Stojmenovic et al (2014) [1] – não foram estudadas no contexto da computação em névoa, mas sim no contexto de redes inteligentes e comunicação máquina-máquina. Existem soluções de segurança para a computação em nuvem sim, mas elas podem não se adequar para a computação em névoa, devido ao fato dos dispositivos funcionarem na borda das redes.

Outra limitação é a grande demanda por transporte de informações. Devido ao enorme e crescente número de dispositivos IoT no mercado, é esperado que estes dispositivos exijam o envio cada vez mais frequente de pequenos blocos de dados (sessões) necessários para atualizar e sincronizar. A frequência das sessões acima referidas causará um grande impacto no atraso da própria rede, e esta parte da infraestrutura deve ser entregue de forma segura para garantir o fluxo de dados seguro.

Na computação em névoa, um grande problema é o ataque man-in-the-middle, pois ele possui um grande potencial para se tornar um ataque típico neste paradigma. Neste ataque, os gateways que servem como dispositivos de Fog podem ser comprometidos ou substituídos por falsos. Exemplos disso são os clientes de determinadas lojas que se conectam a pontos de acesso maliciosos que fornecem SSID enganosos como legítimos. A comunicação privada das vítimas será sequestrada quando os atacantes assumirem o controle dos gateways.
Além disso, este ataque pode ser muito furtivo, já que ele consumirá apenas uma pequena quantidade de recursos nos dispositivos, como a utilização desnecessária da CPU e o consumo de memória. Portanto, os métodos tradicionais de detecção dificilmente podem expor ataques como este. As técnicas de comunicação criptografadas também não podem proteger os usuários deste ataque, uma vez que os invasores podem configurar um terminal legítimo e reproduzir a comunicação sem descriptografá-la. Particularmente, criptografia complexa pode não ser adequada para alguns cenários. Por exemplo, técnicas de criptografia e descriptografia consumirão muita energia da bateria em smartphones.

A convergência de dispositivos que surge a partir do IoT estimula maior demanda de um certo grau de QoS (Quality of Service) esperado da infraestrutura de rede associada. Novos aplicativos móveis que fornecem determinados serviços podem exigir o envio mais frequente de pequenos blocos de dados (sessões) necessários para atualizar e sincronizar. A frequência de sessões

acima referidas terá um grande impacto no atraso e na permeabilidade da própria rede. Esta parte da infraestrutura deve ser entregue de forma segura para garantir o fluxo de dados seguro.

6. POSSÍVEIS SOLUÇÕES

Para proteger os usuários contra riscos de segurança que acompanham o aumento do número de dispositivos IoT interconectados, é evidente que as organizações devam dar mais importância para a detecção e correção de vulnerabilidades em software antes que elas possam ser exploradas por usuários mal-intencionados. Os fabricantes de dispositivos em redes deverão introduzir padrões de segurança mais elevados. Será cada vez mais provável que os dispositivos IoT passem por processos de verificação e certificação de segurança cibernética antes de seu lançamento no mercado. Além disso, soluções de segurança que monitoram o tráfego de rede entre o cliente e o provedor de serviços em nuvem têm demanda cada vez mais alta.

Há vertentes sobre a forma mais efetiva de garantir a segurança em IoT sem comprometer o custo da plataforma e ainda assim assegurar sua expansão. As soluções de segurança não podem trabalhar de forma extremamente pesada em cima dos sensores, a fim de não comprometer o valor desses equipamentos. A segurança em IoT aumenta a complexidade, uma vez que é preciso garantir mais segurança não apenas em suas operações, mas estar atento também às conexões com dispositivos próximos. Na medida em que mais dados são compartilhados, os possíveis pontos de falha crescem e irão exigir soluções mais seguras na própria base.

Dito isso, apresentaremos algumas soluções específicas de segurança - dadas por autores de artigos correlatos - para casos em que foram observados vulnerabilidades nestes dispositivos. Dentre eles destacam-se:

- o problema do ataque man-in-the-middle;
- a detecção de intrusão e anomalias;
- e a aplicação de criptografia sobre os dados que transitam entre estes dispositivos.

Stojmenovic et al (2014) [1] abordam de maneira sucinta um trabalho realizado por outros autores, onde os quais desenvolvem um algoritmo para o monitoramento dos resultados do fluxo de potência e detecção de anomalias nos valores de entrada que poderiam ter sido modificados por ataques. O algoritmo detecta a intrusão usando a análise de componentes principais para separar a variabilidade do fluxo de potência em subespaços regulares e irregulares.

Para o problema do ataque do homem do meio, Stojmenovic et al (2014) [1] explicam que as técnicas tradicionais de detecção de anomalias dependem do desvio das características da comunicação atual e de uma comunicação normal, como consumo de memória, utilização da CPU, uso da largura de banda, etc. Portanto, para detectar o ataque do homem do meio, examina-se o consumo de memória e a utilização da CPU do gateway durante o ataque. Se o ataque não mudar muito as características da comunicação, pode ser provado ser um ataque furtivo. Ou seja, assume-se que o atacante só irá reproduzir os dados em seu próprio computador, mas não irá modificá-los.

Vishwanath et al (2016) [4] sugere a aplicação da técnica de criptografia sobre os dados em névoa para mais segurança. Seu objetivo é obter segurança no segundo nível do sistema de computação em nuvem, fazendo uso do algoritmo de criptografia AES, que é uma criptografia de chave simétrica fazendo uso de uma chave secreta comum para criptografia e descriptografia.

Lu et al (2011) [13] descreveu um esquema de agregação eficiente e que preserva a privacidade para as comunicações de redes inteligentes. Ele usa uma sequência superincreasing para estruturar dados multi-dimensionais e criptografar os dados estruturados pela técnica de criptografia homomórfica. Uma função homomórfica toma como entrada os dados criptografados dos medidores inteligentes e produz uma criptografia do resultado agregado. O dispositivo Fog não pode decifrar as leituras do medidor inteligente e manipulá-las. Isso garante a privacidade dos dados coletados pelos medidores inteligentes, mas não garante que o dispositivo Fog transmita o relatório correto para os outros gateways. Para as comunicações de dados do usuário para o centro de operação da

rede inteligente, a agregação de dados é realizada diretamente no texto criptografado em gateways locais sem descriptografia, e o resultado da agregação dos dados originais pode ser obtido no centro de operação. O custo de autenticação é reduzido por uma técnica de verificação em lote.

Outro problema enfrentado na atualidade é a poluição nas cidades. A poluição causada principalmente pelo tráfego da cidade é um problema em grandes cidades que possui soluções neste contexto. O tráfego contribui igualmente para a redução da qualidade do ar e e para a emissão de gases com efeito estufa. Os engarrafamentos contribuem para o aumento dos custos das atividades económicas e sociais na maioria das cidades. Todos os problemas acima mencionados podem ser reduzidos através da captura e processamento contínuos de dados de tráfego. Em relação ao tráfego através de redes de sensores sem fio de longa distância, o IoT permite monitorar constantemente o tempo de viagem, desde pontos de origem até pontos de destino, de poluição do ar e poluição sonora. Esse tipo de sistema IoT provavelmente substituirá os sistemas atuais de coleta de dados de tráfego e apoiará o desenvolvimento de algoritmos de gerenciamento de tráfego, incluindo sistemas de controle de objetos mais elevados. Os resultados do processamento de dados recolhidos serão apresentados aos passageiros, que terão uma visão contínua da situação do tráfego.

7. PROJETO E DESENVOLVIMENTO DE UMA PROPOSTA

Relacionado ao tema deste artigo, o projeto proposto por Vishwanath et al (2016) [4] tratará sobre a eficácia da criptografia aplicada sobre os dados em Fog. O objetivo deste projeto é obter segurança em Fog, que é o segundo nível do sistema de nuvem, fazendo uso do algoritmo de criptografia AES e aplicando-o sobre os conjuntos de dados selecionados através da implantação em um dispositivo de borda móvel e assim coletar as métricas de desempenho em três conjuntos de dados e avaliar melhores e piores casos em todos os aspectos dos conjuntos de dados.

Para analisá-lo no ambiente de Fog Computing, o dispositivo é considerado e o aplicativo é projetado para criptografar e descriptografar os conjuntos de dados escolhidos usando a técnica de criptografia AES, que é uma criptografia de chave simétrica fazendo uso de uma chave secreta comum para cifragem e decifragem.

Diferentes conjuntos de dados com diferentes tamanhos de dados e de texto, sequências de caracteres e imagens são selecionados para testar o método de criptografia. O desempenho é avaliado para esses conjuntos de dados que são testados implementando no dispositivo. Ao avaliar vários fatores como criptografia, tempo de decodificação, utilização da memória, tempo de resposta para cada conjunto de dados.

7.1. ESPECIFICAÇÃO FORMAL E INFORMAL

Advanced Encryption Standard, que também é conhecido como Rijindeal, é uma técnica de criptografia usada pelo governo dos EUA. AES é conhecida por ter substituições e permutações e é dito ser rápido tanto em software quanto em hardware. Tem tamanho de bloco fixo de 128 bits e tamanho de chave de 128, 192 ou 256 bits. AES opera na ordem principal da coluna 4x4. AES estará executando muitas iterações de transformação para converter o texto original em texto cifrado. Abaixo estão o número de repetições para cada um dos tamanhos da chave de bit.

10 ciclos de repetição para chave de 128 bits
12 ciclos de repetição para chave de 192 bits
14 ciclos de repetição para chave de 256 bits

Estas são as quatro etapas que serão realizadas no conjunto de dados:

1. Etapa SubBytes

O byte de substituição de cada byte pode ser encontrado na tabela de pesquisa. O tamanho da tabela de pesquisa é 16 × 16. O byte de substituição para entrada dada pode ser encontrado dividindo o byte em dois padrões de 4 bits, resultando em um valor inteiro de 0 a 15. Estes podem ser representados por valores Hexadecimais de 0 a F, onde um é usado para encontrar a linha índice e outro é usado para o índice de coluna para consultar na tabela de pesquisa 16 × 16. Na fig 2, cada etapa SubBytes do conjunto de dados é substituída pela tabela de pesquisa de 8 bits. O passo de Substituição concentra-se na redução da correlação entre bits de entrada e saída no nível de bytes.

Algoritmo 1

```
Void SubByte(byte[][] state) {
        for (int rw=0; rw<4: rw++)
        for (int cl=0; cl<N; cl++)
        state[rw][cl]=SBox[state[rw][cl]];
}
```

- **Passo 1:** Como no algoritmo 1, inicialmente o conjunto de dados é armazenado no bloco.
- **Passo 2:** Em seguida, cada um dos blocos será considerado, que tem tamanho de 256 bits.
- **Passo 3:** Agora, cada bloco é dividido em dois e considerado como linha e valor de coluna da caixa S.
- **Passo 4:** Agora o valor é retirado da caixa S e os dados são substituídos pelo valor hexadecimal.
- **Passo 5:** Agora os passos 1-4 são continuados para todos os blocos da mesma maneira

2. Etapa ShiftRows

A representação de matriz mais importante da matriz de estado acontece aqui como na Fig.3. A transformação ShiftRow tem o seguinte comportamento:

1) Não mudará a matriz de estado na primeira linha.
2) Circularmente, a segunda linha será deslocada por um byte para a esquerda.
3) Na terceira linha, deslocando circularmente dois bytes para a esquerda.
4) Na quarta linha, deslocará circularmente três bytes para a esquerda.

Na etapa ShiftRow, cada uma das linhas será trocada para a esquerda dependendo do índice da linha. Do mesmo modo para descriptografar, as linhas correspondentes serão deslocadas para a direção oposta. A primeira linha permanece inalterada, na segunda linha a linha será deslocada para a direita por um byte. A terceira linha será deslocada para a direita por 2 bytes e a quarta linha será deslocada 3 bytes para a direita.

Algoritmo 2

```
Void ShiftRow(byte[ ][ ] state) {
        byte[ ] s= new byte[4];
        for (int t=1; t<4; t++)
                for (int d=0; d<N; d++)
                        s[d]=state[t][(d+t)%N];
        for (int d=0; d<N; d++)
                state[t][d]=s[d];
}
```

- **Passo 1:** no algoritmo 2, os valores hexadecimais serão deslocados para a esquerda, a linha 1 não será deslocada.
- **Passo 2:** Na linha 2, será deslocado para 1 byte para a esquerda, o loop continuará até todos os blocos na linha são deslocados para a esquerda.
- **Passo 3:** na linha 3, o bloco será deslocado para 2 bytes para a esquerda e continuará para todos os bytes na linha.
- **Passo 4:** Na linha 4, o bloco será transferido para a esquerda por 3 bytes e o mesmo processo continuará.

3. Etapa MixColumns

Na coluna de mistura, cada byte da coluna no conjunto de dados é substituído por função de todos os bytes na coluna existente. E, mais importante, cada byte na coluna será substituído por duas vezes esse byte, mais três vezes o próximo byte, além do byte que vem depois, além do byte a seguir.

Algoritmo 3

```
Void MixColumn(byte[ ][ ] st) {
        byte [ ] p= new byte[4];
        for (int cl=0; cl<4; cl++) {
                p[0]=(0x02 # st[0][c]) ^ (0x03 # st[1][c]) ^ st[2][c] ^ st[3][c];
                p[1]= st[0][c] ^ (0x02 # st[1][c]) ^ (0x03 # st[2][c]) ^ st[3][c];
                p[2]= st[0][c] ^ st[1][c] ^ (0x02 # st[2][c]) ^ (0x03 # st[1][c]);
                p[3]=(0x03 # st[0][c]) ^ st[1][c] ^ st[2][c] (0x02 # st[3][c]);

                for( int j=0; j<4; j++)
                        st[i][cl]=p[j];
        }
}
```

- **Passo 1:** como no algoritmo 3, tomamos uma coluna por vez e começamos a aplicar a multiplicação nela.
- **Passo 2:** cada uma das colunas é multiplicada contra o valor da matriz.
- **Passo 3:** agora os resultados serão XOR e gera quatro bytes de resultado para o próximo estado.
- **Passo 4:** agora a multiplicação será aplicada a uma linha de matriz em uma coluna de estado.

4. Etapa AddRoundKey

Na etapa AddRoundkey, cada um dos bytes é combinado com bytes de Roundkey usando a operação XOR.

Algoritmo 4

```
Void AddRoundKey(byte[ ][ ] sta) {
        for (int cl=0; cl<N; cl++)
                for (int rw=0; rw<4; rw++)
                        sta[r][c] = sta[r][c] ^n[nCount++];
}
```

- **Passo 1:** no algoritmo 4, cada um dos 16 bytes no estado será XOR com os 16 bytes da chave expandida para rodada presente.
- **Passo 2:** isto será continuado para todas as linhas no estado.
- **Passo 3:** na próxima rodada de operação AddRoundkey, não chamaremos os primeiros 16 bytes de chave expandida, mas, em vez disso, usamos os bytes de 17 a 32.
- **Passo 4:** da mesma forma que passamos para outras rodadas no estado.

Desta forma, o algoritmo AES executa estes quatro passos de operações em cada uma das iterações e gera um texto cifrado como a Fig. 3. O processo inverso das quatro etapas acima da criptografia será a decodificação em que realizamos todas as quatro etapas acima e as iterações também são realizadas, de modo a gerar o texto simples.

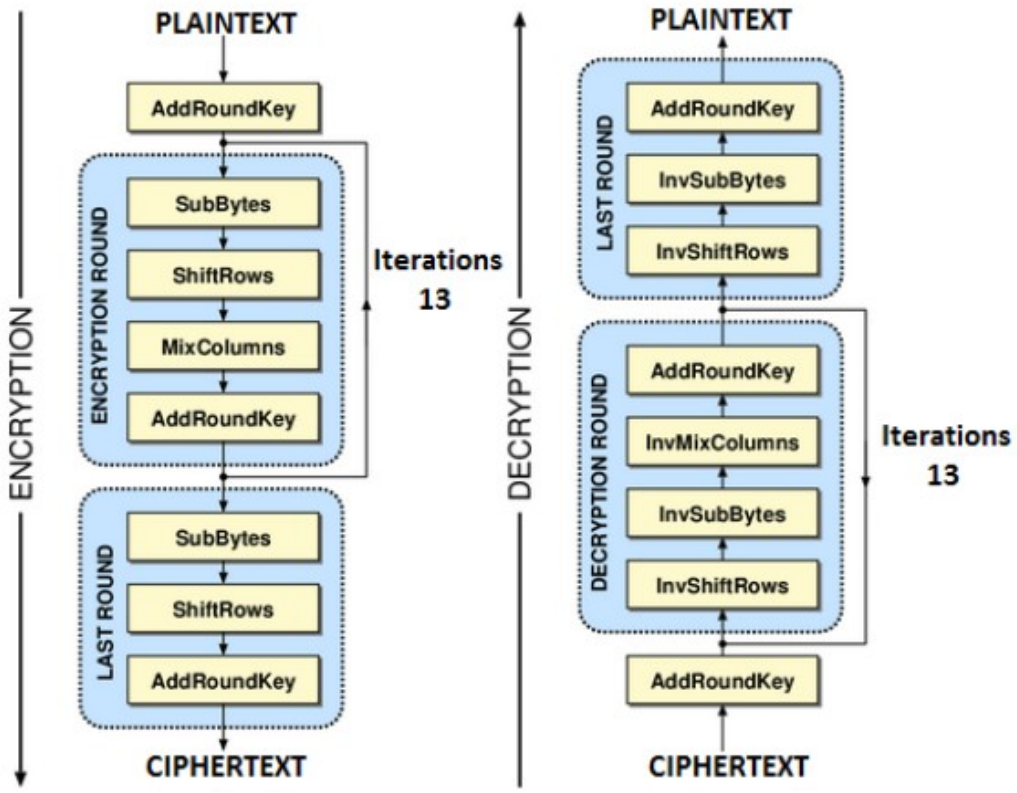


Fig.3. Arquitetura do algoritmo AES

7.2. IMPLEMENTAÇÃO

Para demonstrar a AES, nós o implementamos no dispositivo final com o Android como sistema operacional. Nos quais os arquivos serão selecionados e o usuário precisa dar o caminho do arquivo e fornecer uma senha, que atuará como chave e a mesma chave precisa ser inserida enquanto desencripta os dados.

Fig.4. Interface de usuário da aplicação

A interface de usuário geral da aplicação é projetada com opções como na Fig. 4 para carregar o arquivo do cartão de memória do dispositivo e, em seguida, a senha deve ser dada para criptografar o conjunto de dados selecionado e o botão criptografar permite que o conjunto de dados seja criptografado. Para criptografar os dados, ele estará usando o método *encrypt(byte[] in, byte[] key)* que irá converter o texto simples no texto cifrado.

- *generatekey(byte[] key)* é o método usado para geração da chave.
- *encryptblk(byte[] blk)* é o método usado para criptografar todo o bloco de dados.
- O método acima também usa os métodos *SubByte()*, *ShiftRow()*, *MixcColumn(),* e *AddRoundKey()* em cada uma das iterações para produzir um texto cifrado.
- Na operação geral, a saída gerada estará no texto cifrado no formato de byte[].

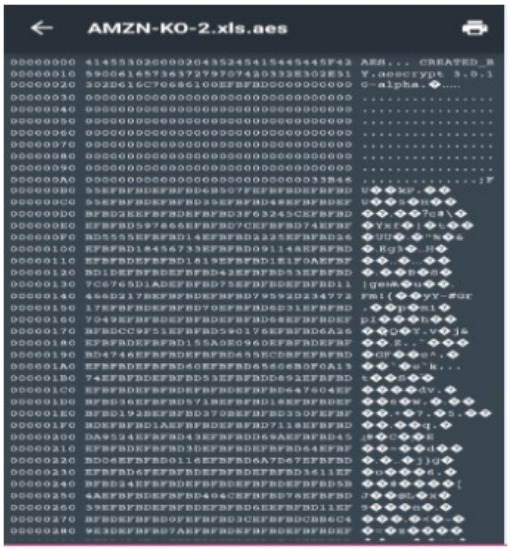
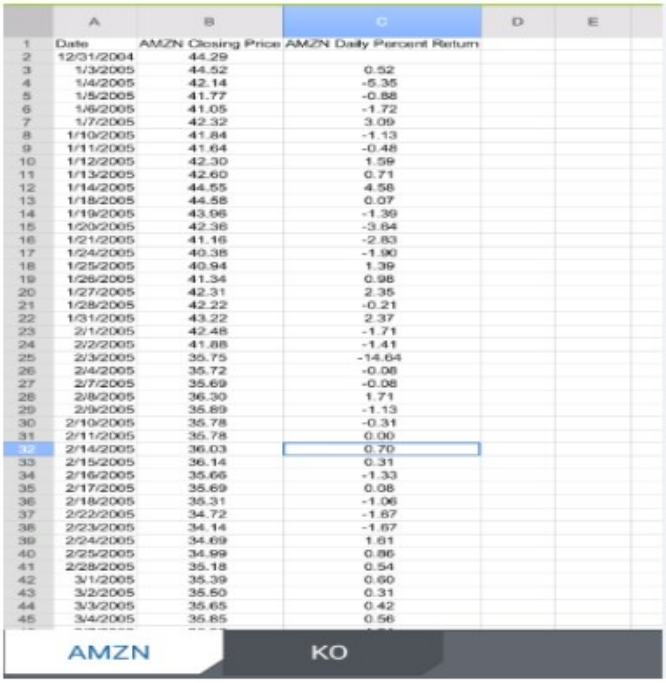


Fig.5. Conjunto de dados criptografados da Amazon-Coca-Cola

A Fig 5 mostra o arquivo criptografado que é gerado na criptografia do arquivo do conjunto de dados Amazon e Coca-Cola. Ele mostra como o padrão é usado e cada linha dos dados é criptografada. Da mesma forma, para a descriptografia, ele usa o método *decrypt(byte[] in, byte[] key)* para converter o texto cifrado no texto original.

- *generatekey(byte[] key)* é o método para geração da chave.
- *decryptblk(byte[] blk)* é o método usado para decifrar todo o texto cifrado.
- Os métodos acima usam *RevSubBytes()*, *RevShiftRows()*, *RevMixcolumns(),* e *RevAddRoundkey()* para decifrar o arquivo de dados.
- A saída gerada estará no formato de byte[].



	A	B	C	D	E
1	Date	AMZN Closing Price	AMZN Daily Percent Return		
2	12/31/2004	44.29			
3	1/3/2005	44.52	0.52		
4	1/4/2005	42.14	-5.35		
5	1/5/2005	41.77	-0.88		
6	1/6/2005	41.05	-1.72		
7	1/7/2005	42.32	3.09		
8	1/10/2005	41.84	-1.13		
9	1/11/2005	41.64	-0.48		
10	1/12/2005	42.30	1.59		
11	1/13/2005	42.60	0.71		
12	1/14/2005	44.55	4.58		
13	1/18/2005	44.58	0.07		
14	1/19/2005	43.96	-1.39		
15	1/20/2005	42.36	-3.64		
16	1/21/2005	41.16	-2.83		
17	1/24/2005	40.38	-1.90		
18	1/25/2005	40.94	1.39		
19	1/26/2005	41.34	0.98		
20	1/27/2005	42.31	2.35		
21	1/28/2005	42.22	-0.21		
22	1/31/2005	43.22	2.37		
23	2/1/2005	42.48	-1.71		
24	2/2/2005	41.88	-1.41		
25	2/3/2005	35.75	-14.64		
26	2/4/2005	35.72	-0.08		
27	2/7/2005	35.69	-0.08		
28	2/8/2005	36.30	1.71		
29	2/9/2005	35.89	-1.13		
30	2/10/2005	35.78	-0.31		
31	2/11/2005	35.78	0.00		
32	2/14/2005	36.03	0.70		
33	2/15/2005	36.14	0.31		
34	2/16/2005	35.66	-1.33		
35	2/17/2005	35.69	0.08		
36	2/18/2005	35.31	-1.06		
37	2/22/2005	34.72	-1.67		
38	2/23/2005	34.14	-1.67		
39	2/24/2005	34.69	1.61		
40	2/25/2005	34.99	0.86		
41	2/28/2005	35.18	0.54		
42	3/1/2005	35.39	0.60		
43	3/2/2005	35.50	0.31		
44	3/3/2005	35.65	0.42		
45	3/4/2005	35.85	0.56		

Fig.6. Arquivo decifrado.

A Fig. 6 mostra o arquivo que foi descriptografado após a criptografia do conjunto de dados escolhido. Isso revela o arquivo original do conjunto de dados do Amazon-Coca-Cola ao selecionar a opção decrypt e carregar o arquivo criptografado. Da mesma forma, o algoritmo foi implementado em outros dois conjuntos de dados e observamos que os dados estão sendo criptografados e também descriptografados.

7.3. VERIFICAÇÃO, VALIDAÇÃO E TESTES

Para implementar este processo de criptografia sobre os dados, escolhemos três conjuntos de dados usando algumas fontes que contêm diferentes tipos de dados de diferentes tamanhos. Os conjuntos de dados que foram considerados para implementar a criptografia são:

- **Amazon.com & Coca-cola:** Retornos diários, por dez anos (2005 a 2014) para os estoques de duas empresas. Este conjunto de dados é escolhido para indicar o desempenho em pequenos dados, pois é coleção de informações de duas grandes empresas Amazon e Coca-cola que possuem o mesmo tipo de dados com conteúdo apenas de números. O tamanho do conjunto de dados é 500kb e contém dados estruturados.

- **US Hospital Charge Data:** Contém informações sobre serviços para pacientes internados e ambulatoriais. Este conjunto de dados é escolhido para indicar desempenho em dados grandes, pois tem muitos tipos diferentes de campos com números, strings, caracteres especiais e de tamanho enorme. O tamanho deste conjunto de dados é de cerca de 5 Mb e contém o nome, detalhes da pessoa. Isso contém mais de 150000 linhas nele.

- **Estatísticas de Crimes:** Este arquivo contém informações sobre todas as fraudes e crimes recentes nos últimos tempos. Este conjunto de dados é escolhido para indicar desempenho em dados de maior tamanho e diferentes tipos de conteúdo como imagens, representações pictóricas, strings, números. Este é um conjunto de dados não estruturado, que tem um tamanho de quase 10 MB.

A) Configurações de Simulação

Agora, a implementação da criptografia AES sobre esses conjuntos de dados é realizada, onde todos os dados dos conjuntos de dados são criptografados e também são descriptografados sem perda de dados. Então, para executar esse processo sobre a camada do sistema da nuvem, o dispositivo Android é considerado e o algoritmo foi implantado no celular. Ambiente para o dispositivo de borda:

- Ferramenta de Desenvolvimento: Android SDK
- Linguagem de Programação: JAVA
- IDE: Intellij
- Banco de Dados: MySQL

Para testar, o Apk está instalado no One plus One, que é um cyanogen Android OS 12. Isso usa o EMMC 5.0 para acessar e gravar memória flash de 16GB ou 64GB. 3 GB de RAM LP-DDR3. Processador Qualcomm .801 com CPUs Quad-Core de 2,5 GHz.

As métricas que são comparadas com os conjuntos de dados por criptografia são:

1. Carga de usuário vs Tempo de CPU

Considera como o tempo de CPU varia quando AES está sendo usado em diferentes tamanhos de conjuntos de dados. Então, ele retorna o tempo de CPU para cada tamanho de conjuntos de dados que foram selecionados.

2. Tamanho do arquivo vs Tempo de criptografia

É usado para calcular o tempo necessário para criptografia no caso de cada conjunto de dados de diferentes tamanhos.

3. Tamanho do arquivo vs Tempo de descriptografia

É usado para calcular o tempo necessário para descriptografar no caso de cada conjunto de dados de diferentes tamanhos.

4. Tamanho do arquivo vs Utilização da memória

Retorna resultados de quanta memória será utilizada no uso de diferentes tamanhos de conjuntos de dados. Assim, retorna a utilização da memória para cada tamanho de conjuntos de dados que foram selecionados.

A precisão também é determinada para cada conjunto de dados para verificar se o conjunto

de dados completo está criptografado e descriptografado também após o descriptografado do conjunto de dados sem perda de dados. Para comparar e avaliar o desempenho dos conjuntos de dados por criptografia, uma ferramenta do Simulink é usada executando-o com a ferramenta de análise linear. Nesse caso, o código java será executado na ferramenta de software Simulink e o gráfico será traçado de acordo, considerando criptografia e descriptografia, considerando as métricas fornecidas.

Os dados para as métricas são coletados usando técnicas de processamento em lote e funções de *getMemoryinfo()* para utilização de memória e *getCpuStat()* para o tempo gasto pela CPU e *timestamp()* para criptografia e tempo de descriptografia são usados nos scripts java e os gráficos são plotados em conformidade em Ferramenta de Simulink.

B) Resultados da Simulação

1. Carga de usuário vs Tempo da CPU

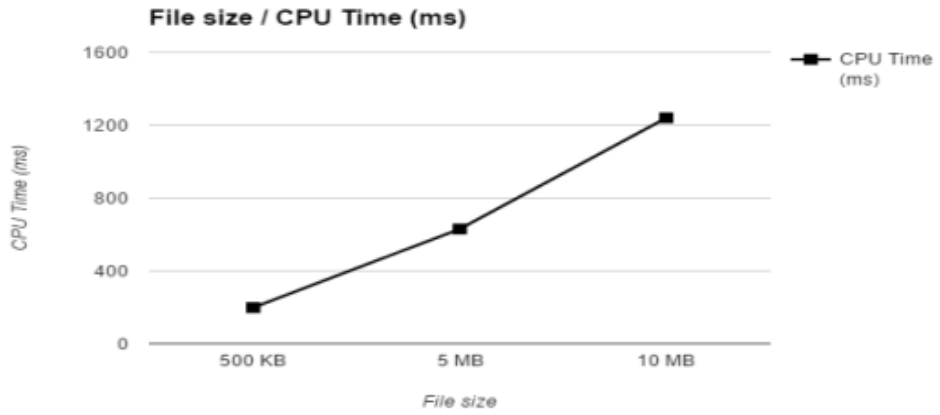


Fig.7. Comparando o tempo gasto pela CPU para cada tamanho do conjunto de dados escolhido.

Fig 7 mostra como o tempo de CPU varia para cada tamanho dos conjuntos de dados: 500Kb, 5Mb e 10Mb.

2. Tamanho de arquivo vs Tempo de criptografia

Fig.8. Comparando o tempo necessário para criptografar conjunto de dados para cada um dos tamanhos dos conjuntos de dados.

A Fig. 8 mostra como o tempo de criptografia está variando no caso de cada conjunto de dados, dependendo do seu tamanho

3. Tamanho do arquivo vs Tempo de descriptografia

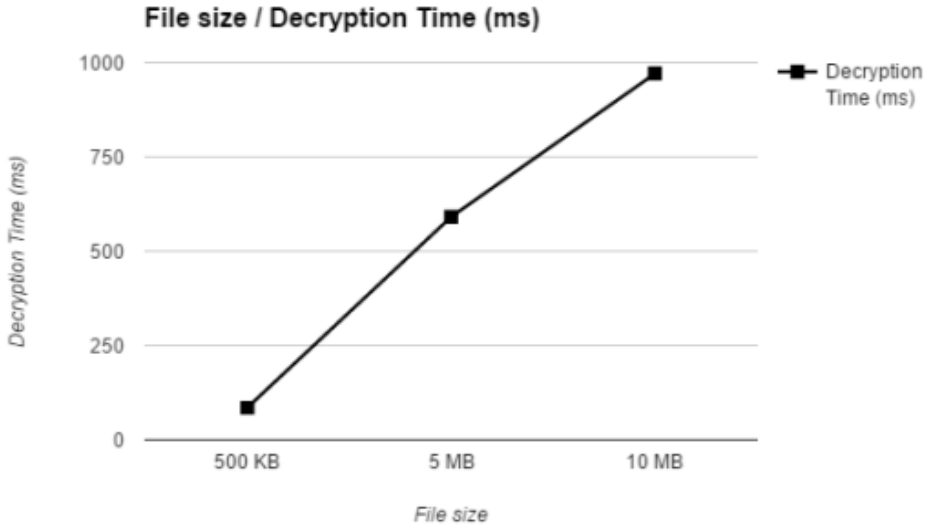


Fig.9. Comparando o tempo de decodificação do conjunto de dados para cada um dos tamanhos de conjunto de dados.

A Fig. 9 fornece como o tempo necessário para o decodificação muda à medida que o tamanho do conjunto de dados é variado.

Observação: Tanto o tempo de criptografia quanto o tempo de descriptografia não são iguais no caso de cada conjunto de dados. Essa alteração pode ser assumida porque, enquanto estamos criptografando os dados, cada um dos blocos deve ser criptografado sequencialmente e enquanto que no descriptografia podemos aplicar a operação XOR em todos os blocos paralelamente.

4. Tamanho do arquivo vs Utilização da memória

Fig.10. Comparando a memória utilizada para cada conjunto de dados de diferentes tamanhos.

A Fig. 10 dá a quantidade de memória que foi utilizada para cada conjunto de dados de diferentes tamanhos.

Avaliação de cada conjunto de dados
Outro dispositivo de borda - o laptop é considerado para avaliar os melhores e piores casos para cada um dos conjuntos de dados.
Possui processor i5, 8GB RAM, CPU @ 2.4GHz.

Conjunto de Dados 1 - 500KB

Tabela 1

Dispositivo	Tempo de CPU (ms)	Tempo de Codificação (ms)	Utilização da Memória (MB)
Laptop	91	87	7.5
Mobile	98	90	7.5

Na tabela 1, os valores para o conjunto de dados de 500Kb em relação a cada métrica em dois dispositivos de borda são fornecidos.

Melhor caso: melhor caso no tempo de criptografia e tempo de CPU.

Conjunto de Dados 2 – 5MB

Tabela 2

Dispositivo	Tempo de CPU (ms)	Tempo de Codificação (ms)	Utilização da Memória (MB)
Laptop	426	436	32
Mobile	630	626	32

Na tabela 2, os valores para o conjunto de dados que é de 5mb em relação a cada métrica em dois dispositivos de borda são fornecidos.

Melhor caso: no tempo de criptografia.
Pior caso: no tempo da CPU.

Conjunto de Dados 3 – 10MB

Tabela 3

Dispositivo	Tempo de CPU (ms)	Tempo de Codificação (ms)	Utilização da Memória (MB)
Laptop	1103	973	53
Mobile	1240	1023	53

Na tabela 3, valores para o conjunto de dados que é de 10MB em relação a cada métrica em dois dispositivos de borda são fornecidos.
Pior caso: no tempo de criptografia e no tempo de CPU.

A utilização da memória através da AES no celular pode ser considerada como um caso médio.

7.4. ANÁLISE

O desempenho é avaliado através dos conjuntos de dados usando diferentes métricas. Quando cada conjunto de dados é considerado em dispositivos móveis e laptop, há muito espaço de tempo para a CPU no caso de conjuntos de dados maiores de 5MB e 10MB, mas, no caso de conjuntos de dados menores, o tempo de CPU é quase igual comparado em ambos os dispositivos. Isso pode transmitir que o celular pode ter uma boa utilização do tempo de CPU para conjuntos de dados menores em vez de conjuntos de dados maiores. Ao chegar à utilização da memória, será o mesmo para qualquer dispositivo com RAM. O tempo de criptografia para os conjuntos de dados é menor no laptop quando comparado ao celular que pode ser devido à diferença na velocidade dos processadores. No que diz respeito aos conjuntos de dados, quando utilizados em dois dispositivos diferentes, ele mostra os melhores resultados em um dispositivo de alto processamento. Mas como Fog será tomada em cada dispositivo pequeno e como o celular está tendo a melhor adaptabilidade para a segurança para conjuntos de dados menores, ele fornece uma boa criptografia e, portanto, segurança em um dispositivo de ponta. Isso pode mostrar que o Fog pode ser protegido e o sistema em nuvem pode ser fornecido com a segunda camada de proteção através da criptografia em nevoeiro.

8. CONCLUSÕES

Os principais desafios para a realização da Internet of Things e do paradigma de Fog Computing incluem a segurança, a privacidade e a confidencialidade, a gestão das heterogeneidades, as limitações das capacidades de rede, a gestão e o processamento de grandes quantidades de dados, a fim de fornecer informações/serviços úteis e permitir uma política regularmente eficiente. Aliando tudo isso ao fato de que ambos os paradigmas são extremamente importantes e necessários em nossa sociedade, torna-se evidente que atitudes devem ser tomadas, com o intuito de resolver suas deficiências e permitir sua evolução.

Conforme foi visto anteriormente, os pilares centrais da segurança da informação são a *confidencialidade*, que é o ato de manter a informação indisponível para indivíduos, entidades ou processos não-autorizados; a *integridade*, que garante que a informação manipulada mantenha todas as características originais estabelecidas pelo proprietário da informação; a *disponibilidade*, que garante que a informação esteja sempre disponível para usuários legítimos; a *autenticidade*, que é a propriedade que garante que a informação veio de uma fonte confiável e que não foi alvo de mutações ao longo de um processo; a *irretratabilidade* ou *não-repúdio*, que garante a impossibilidade de negar a autoria em relação a uma transação anteriormente feita; e a *conformidade*, que garante que o sistema deve seguir as leis e regulamentos vinculados a este tipo de processo. Resumindo, para manter a segurança da informação intacta, a implementação destes conceitos deverá ser realizada de maneira correta e minuciosa, evitando, assim, brechas em implementações e protocolos, e reduzindo a probabilidade de eventuais ataques.

Ameaças à segurança constituem-se de perda de confidencialidade, integridade e/ou disponibilidade. Para impedir o extravio destas propriedades, diferentes técnicas podem ser utilizadas, tais como aplicações IDS (Intrusion Detection System), que são sistemas que conseguem descobrir determinados tipos de ataques em uma rede pesquisando-se por acessos não autorizados, criptografia na comunicação de rede entre os vários sistemas interligados, protocolos de rede que não possuam brechas em suas implementações, dentre várias outras vistas até aqui. Com isso, queremos destacar que a segurança depende de vários fatores, e sempre há o surgimento de novas brechas e a localização de brechas que já existiam há muito tempo nos sistemas, mas que ainda não haviam sido exploradas.

Estima-se que, até 2020, existirão 34 bilhões de dispositivos conectados à Internet no mundo, contando com PCs, smartphones, tablets, smartTVs, relógios inteligentes e dispositivos IoT. São mais de quatro dispositivos para cada ser humano no planeta. Esse crescimento exponencial requer um crescimento e evolução de todas as áreas que os envolvem. Não será um caminho fácil a

se percorrer, mas será ainda pior caso não estejamos preparados quando o momento chegar. Toda a infraestrutura da rede, segurança e energia deverá ser capaz de suprir as necessidades dessa demanda, e ela deverá continuar a crescer. Diversos estudos estão sendo realizados com objetivos de buscar o fortalecimento destas estruturas, otimização em larga escala dos serviços e investir mais do que é investido atualmente em segurança digital, pois muito em breve estaremos totalmente dependentes desta tecnologia, e erros poderão ser catastróficos.

Organizações e fabricantes de dispositivos IoT deverão repensar o modo sobre como estão desenvolvendo e implementando seus produtos, pois estão focando mais em produtos comerciais do que em produtos tecnologicamente seguros, e com isso problemas de falhas de segurança e bugs irão comprometer a segurança destes dispositivos. Pesquisas recentes mostram que uma grande maioria dos dispositivos já estão vindo de fábrica com um alto grau de vulnerabilidade, apresentando, em média, 25 vulnerabilidades diferentes por dispositivo, o que, por si só, traz um alto risco ao usuário. Como falhas de segurança recorrentes destes dispositivos, podemos citar a falta de privacidade com o usuário, pois o sistema recolhe ao menos uma informação pessoal e confidencial do mesmo, falta de criptografia da comunicação de dados com a Internet, além do fato de muitos dispositivos permitirem a criação e configuração de senhas com baixa ou nenhuma complexidade, tais como “1234” ou “ABCD”. Problemas de segurança não envolvem apenas aplicações técnicas, mas também o fator humano, e isso deve ser levado em consideração no planejamento da segurança.

Finalmente, é muito conhecido o uso de aplicativos móveis que são instalados em smartphones para qualquer tipo de gerenciamento, obter dados ou controlar o dispositivo. Como resultado, os aplicativos móveis também podem ser alvos de ataques, explorando vulnerabilidades ou deficiências na sua implementação ou desenvolvendo aplicativos maliciosos que emulam o comportamento e o aparecimento de acesso legítimo aos dispositivos IoT.

Enfim, a segurança é primordial em toda a tecnologia que nos cerca, e ela está e estará muito mais evidente no uso exponencial de dispositivos IoT em praticamente qualquer atividade. Porém, quanto maior o número de dispositivos interligados, maior a chance de infecções nesta rede. Um dispositivo vulnerável já é suficiente para colocar os outros em perigo, por isso a segurança é tão importante neste contexto. Ataques recentes nos mostram que não devemos pensá-la como uma simples ferramenta virtual, mas sim como se fosse a nossa própria segurança física.

9. TRABALHOS FUTUROS

Stojmenovic et al (2014) [1] pretendem expandir o paradigma de computação em névoa em Smart Grid. Onde, neste cenário, dois modelos para dispositivos de névoa poderão ser desenvolvidos: os dispositivos independentes que consultam diretamente a nuvem para atualizações periódicas de preços e demandas; enquanto dispositivos da névoa interligados podem consultar uns aos outros e criar coalizões para aprimoramentos adicionais. Em seguida, a computação baseada em Fog SDN nas redes de veículos receberá a devida atenção. O controle de luz de trânsito também pode ser auxiliado pelo conceito de computação de Fog. Finalmente, a mobilidade entre nós da névoa, e entre névoa e nuvem, pode ser investigada. Ao contrário dos centros de dados tradicionais, os dispositivos Fog são geograficamente distribuídos em plataformas heterogêneas. A mobilidade de serviços em todas as plataformas precisa ser otimizada.

Trabalhos futuros relacionados a computação em névoa também deverão focar em resolver o problema do ataque man-in-the-middle.

Weber et al (2016) [3] citam outros desafios relacionados a IoT que devem receber atenção futuramente, sendo eles:

- a regulamentação do mercado;
- a concepção de uma arquitetura mais eficiente para a ligação em rede dos sensores e o armazenamento dos dados recolhidos;

- o desenvolvimento de mecanismos para o processamento do fluxo de dados recolhidos em redes sensoriais;

- transição para IPv6 (grande número de endereços, possibilidade de configuração automática e parâmetros de segurança melhorados);

- fontes de alimentação de dispositivos/sensores (os dispositivos são alimentados por eletricidade produzidos a partir do ambiente, tais como vibrações, luz e fluxo de ar);

- redução do custo dos componentes IoT.

Para Atzori et al (2014) [14], a tendência atual, que fora destacada em seu artigo, é a de atribuir um endereço IPv6 a cada dispositivo IoT, de modo a permitir alcançá-los a partir de qualquer outro nó da rede. Portanto, é possível que a evolução da Internet exigirá uma mudança nesta tendência.

Outro paradigma interessante que está emergindo no contexto Internet do Futuro é o chamado Web Squared, que é uma evolução da Web 2.0. Destina-se a integrar as tecnologias de web e detecção de forma a enriquecer o conteúdo fornecido aos usuários. Isto é obtido levando em consideração as informações sobre o contexto do usuário coletadas pelos sensores (microfone, câmeras, GPS, etc.) implantados nos terminais do usuário. Nesta perspectiva, observe que o Web Squared pode ser considerado como uma das aplicações que se extendem no contexto de IoT tal como a Web é hoje uma importante aplicação que está sendo executada pela Internet.

Em seu artigo, Hong et al (2013) [15] discutem o design do Mobile Fog, que é um modelo de programação para aplicações em grande escala e sensíveis à latência na Internet das Coisas. Uma vez que esta é uma pesquisa em andamento, surgem alguns problemas de pesquisa interessantes como trabalhos futuros:

- *Implementação do sistema de tempo de execução:* planeja-se desenvolver um sistema de tempo de execução que implemente o modelo de programação Mobile Fog em dispositivos verdadeiramente habilitados para Fog Computing. O principal desafio é desenvolver um sistema de tempo de execução distribuído que possa migrar os processos do Mobile Fog em diferentes dispositivos, ao mesmo tempo que fornece confiabilidade, segurança e isolamento de desempenho para recursos de infraestrutura compartilhada.

- *Algoritmo de escalonamento de processo:* para obter uma melhor utilização da largura de banda da rede, latência e equilíbrio de carga em dispositivos distribuídos, eles estão trabalhando atualmente em um algoritmo que pode encontrar de forma adaptativa um escalonamento quase ótima dos processos do Mobile Fog. O principal desafio nesta direção é encontrar um melhor escalonamento com base em restrições dinâmicas, incluindo recursos disponíveis, carga de trabalho da aplicação e custo de migração para os processos do Mobile Fog.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] STOJMENOVIC, Ivan *et al.***The Fog Computing Paradigm: Scenarios and Security Issues.** 2014 Federated Conference on Computer Science and Information (FedCSIS). IEEE Xplore. Disponível em: <http://ieeexplore.ieee.org/abstract/document/6932989/>. Acesso em: mar 2017.

[2] GAONA-GARCÍA, Paulo *et al.* **Analysis of Security Mechanisms Based on Clusters IoT Enviroments.** 2017. International Journal of Interactive Multimedia and Artificial Intelligence. Disponível em: <http://www.ijimai.org/journal/sites/default/files/files/2016/08/ijimai20174_3_8_pdf_20224.pdf>. Acesso em: mar 2017.

[3] WEBER, Mario *et al.* **Security challenges of the Internet of Things.** 2016. 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE Xplore. Disponível em: <http://ieeexplore.ieee.org/abstract/document/7522219/>. Acesso em: mar 2017.

[4] VISHWANATH, Akhilesh *et al.* **Security in Fog Computing through Encryption.** 2016. International Journal of Information Technology and Computer Science, 2016, 5, 28-36. Disponível em: <http://www.mecs-press.org/ijitcs/ijitcs-v8-n5/v8n5-3.html>. Acesso em: mar 2017.

[5] SÁNCHEZ-ARIAS, G. *et al.* **Midgar: Study of communications security among Smart Objects using a platform of heterogeneous devices for the Internet of Things.** 2017. Publicado em: Future Generation Computer Systems. Elsevier. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167739X17301632>. Acesso em: mar 2017.

[6] SICARI, Sabrina *et al.* **Security, privacy and trust in Internet of Things: The road ahead.** 2015. Computer Networks. Elsevier. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1389128614003971>. Acesso em: mar 2017.

[7] ALMADHOR, Ahmad. **A Fog Computing based Smart Grid Cloud Data Security.** 2016. International Journal of Applied Information Systems (IJAIS). Disponível em: <http://www.ijais.org/archives/volume10/number6/868-2016451515> Acesso em: mar 2017.

[8] DHANDE, Neha Shrikant. **FOG COMPUTING: REVIEW OF PRIVACY AND SECURITY ISSUES.** 2015. International Journal of Engineering Research and General Science. Volume 3, Issue 2, March-April, 2015. Disponível em: <http://pnrsolution.org/Datacenter/Vol3/Issue2/122.pdf>. Acesso em: mar 2017.

[9] SARAF, Kundankumar Rameshwar *et al*. **Text and Image Encryption Decryption Using Advanced Encryption Standard.** 2014. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS). Disponível em: <http://ijettcs.org/Volume3Issue3/IJETTCS-2014-06-11-081.pdf>. Acesso em: jun 2017.

[10] GUBBI, Jayavardhana *et al*. **Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions.** Future Generation Computer Systems, 2013. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167739X13000241>. Acesso em: jun 2017.

[11] PAVITHRA, D. *et al*. **IoT based monitoring and control system for home automation.** 2015. Global Conference on Communication Technologies (GCCT), 2015, pp. 169-173. Disponível em: <http://ieeexplore.ieee.org/abstract/document/7342646/>. Acesso em: jun 2017.

[12] GRANJAL, Jorge *et al*. **Security for the Internet of Things: A Surbey of Existing Protocols and Open Research Issues.** 2015. Communications Surveys & Tutorials, IEEE, vol. 17, pp. 1294-1312, 2015. Disponível em: <http://ieeexplore.ieee.org/abstract/document/7005393/>. Acesso em: jun 2017.

[13] LU, Rongxing *et al.* **GRS: The green, reliability, and security of emerging machine to machine communications.** Communications Magazine, IEEE, vol. 49, no. 4, pp. 28-35, April 2011. Disponível em: <http://ieeexplore.ieee.org/abstract/document/5741143/>. Acesso em: jun 2017.

[14] ATZORI, Luigi *et al*. **The Internet of Things: A survey.** Computer Networks, Volume 54, Issue 15. Elsevier. 2010. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1389128610001568>. Acesso em: jun 2017.

[15] HONG, Kirar *et al*. **Mobile fog: a programming model for large-scale applications on the internet of things.** MCC '13 Proceedings of the second ACM SIGCOMM workshop on Mobile cloud computing. Pages 15-20. 2013. Disponível em: <http://dl.acm.org/citation.cfm?id=2491270>. Acesso em: jun 2017.

[16] **Internet of things research study.** Study led by Hewlett Packard Entreprise Development LP. 2015. Disponível em: <http://h20195.www2.hpe.com/V4/GetDocument.aspx?docname=4AA5-4759ENW>. Acesso em: jun 2017.