**RAFAEL CRUZ RIBEIRO**

# SEGURANÇA E PRIVACIDADE PARA SMART HOUSE UTILIZANDO MOBILE CLOUD COMPUTING

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

**CURITIBA**

**2015**

**RAFAEL CRUZ RIBEIRO**

# SEGURANÇA E PRIVACIDADE PARA SMART HOUSE UTILIZANDO MOBILE CLOUD COMPUTING

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Área de Concentração: *Ciência da Computação*

Orientador: Prof. Dr. Altair O. Santin

**CURITIBA**

**2015**

# Ficha Catalográfica

Ribeiro, Rafael Cruz
R484s Segurança e privacidade para *smart house* utilizando *mobile cloud*
2015 *computing* / Rafael Cruz Ribeiro; orientador, Altair O. Santin. -- 2015
54 f. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2015
Bibliografia: f.49-54

1. Informática. 2. Bancos de dados – Medidas de segurança. 3. Internet - Medidas de segurança. 4. Computação em nuvem. 5. Sistemas de comunicação sem fio. I. Santin, Altair O. II. . Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática. III. Título.

CDD 20. ed. – 004.068

# Ata da Defesa

Pontifícia Universidade Católica do Paraná
Escola Politécnica
Programa de Pós-Graduação em Informática

PUCPR
GRUPO MARISTA

ATA DE DEFESA DE DISSERTAÇÃO DE MESTRADO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

DEFESA DE DISSERTAÇÃO DE MESTRADO Nº 08/2015

Aos 30 dias do mês de Setembro de 2015 realizou-se a sessão pública de Defesa da Dissertação **"Segurança e Privacidade para *Smart House* Utilizando *Mobile Cloud Computing*"** apresentado pelo aluno **Rafael Cruz Ribeiro,** como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Altair Olivo Santin
PUCPR (Orientador)
*(assinatura)* Aprov. *(Aprov/Reprov)*

Prof. Dr. Raphael Carlos Santos Machado
INMETRO
*(assinatura)* APROV *(Aprov/Reprov)*

Prof. Dr. Luiz Fernando Rust da Costa Carmo
INMETRO/UFRJ
*(assinatura)* APROV *(Aprov/Reprov)*

Conforme as normas regimentais do PPGIa e da PUCPR, o trabalho apresentado foi considerado Aprovado (*aprovado/reprovado*), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora registradas no Livro de Defesas do programa.

Programa de Pós-Graduação em Informática PPGIa PUCPR

Prof.ª Dr.ª Andreia Malucelli.
Coordenadora do Programa de Pós-Graduação em Informática.

Rua Imaculada Conceição, 1155 Prado Velho CEP 80215 901 Curitiba Paraná Brasil
Tel.: (41) 3271 1669 Fax.: (41) 3271 2121 www.ppgia.pucpr.br

Dedico este trabalho aos meus descendentes, que em um futuro próximo se beneficiarão da comodidade das smart houses de maneira segura e privada.

# Agradecimentos

À Deus, por me oferecer tudo o que eu preciso.

À minha família, em especial mamãe, vovó e Elcio, por não medir esforços para que esse sonho se realizasse.

À minha namorada Fernanda Guidi Fabris, pela imensa ajuda, compreensão e amor.

Ao professor Dr. Altair Olivo Santin, pela sua dedicação, sabedoria, ensinamentos. Agradeço principalmente pela paciência e pelo voto de confiança.

Ao meu colega Arlindo Marcon Júnior, por me proporcionar essa grande oportunidade de cursar esse mestrado.

Aos meus amigos do SecPLab. Em especial aos pesquisadores Cleverton, Eduardo e Vilmar pela grande ajuda técnica na elaboração deste trabalho e do artigo, além dos incentivos diários. Agradeço também pelos momentos de descontração e lazer que tornaram a realização deste trabalho menos cansativo. Agradeço também ao Adriano pelas palavras de incentivo nos momentos difíceis.

# Sumário

# Lista de Figuras

# Lista de Tabelas

# Lista de Abreviaturas

**AMI** *Advanced Meter Infrastructure*

**HAN** *Home-Area Network*

**IAA** *I am Alive*

**IaaS** *Infrastructure as a Service*

**IdM** *Identity Management*

**IMEI** *International Mobile Equipment Identity*

**IoT** *Internet of Things*

**IP** *Internet Protocol*

**LAN** *Local-Area Network*

**MAC** *Media Access Control*

**MCC** *Mobile Cloud Computing*

**NAN** *Neighborhood-Area Network*

**NIST** *National Institute of Standards and Technology*

**P2P** *Peer-to-Peer*

**PaaS** *Platform As a Service*

**PAN** *Personal-Area Network*

**PLC** *Power Line Communication*

**SaaS** *Software as a Service*

**SCADA** *Supervisory Control and Data Acquisition*

**SMS** *Short Message Service*

**SSO** *Single Sign-On*

**WAN** *Wide-Area Network*

# Resumo

Uma *smart grid* pode interagir com uma *smart house* para solicitar reconfigurações nos perfis de consumo, objetivando a continuidade do fornecimento de energia. Os *appliances* da *smart house* não podem estar acessíveis diretamente a partir da Internet para evitar que um atacante controle remotamente os *appliances* ou viole a privacidade dos moradores, monitorando seus hábitos de consumo para inferir que estão ausentes, por exemplo. Este trabalho objetiva prover segurança e privacidade *by design* em *smart house*, isolando os *appliances* da Internet, mas deixando algumas funcionalidades relacionadas ao consumo disponíveis externamente. A coleta de dados de consumo, processamento e alterações nos perfis de consumo são possíveis apenas internamente a *smart house*, a partir de uma nuvem computacional formada de dispositivos móveis (MCC - *Mobile Cloud Computing*). Assim, é possível que a companhia de energia requisite a reconfiguração do perfil de consumo da *smart house* em função de limitação de fornecimento de energia, por exemplo. O protótipo e os testes mostram que nossa proposta é factível, realizando a coleta, processamento e armazenamento de 1200 atributos dos *appliances* em 15,25 segundos, utilizando 03 dispositivos móveis na MCC.

**Palavras-Chave**: *Smart House,* Segurança, Privacidade, *Mobile Cloud Computing*.

# Abstract

Smart house allow residents to access appliances and the smart grid to reconfigure its power consumption. This paper aims to provide security and privacy by design in a smart house, isolating appliances from direct Internet access, however allowing external entities to access power consumption features. Data acquisition, processing, and consumer profiles updates are only possible inside a smart house, using a Mobile Cloud Computing (MCC). Moreover, an electric power company can request the consumption profile reconfiguration of smart houses, for instance to avoid a potential blackout. Thus, it is expected an intruder cannot control the appliances remotely, neither violate the residents privacy, monitoring their power consumption habits, in order to infer the house is alone. The prototype tests showed that our proposal is feasible, allowing to collect, process and store 1200 appliances attributes in 15.25 seconds while using only three mobile devices in the MCC.

**Keywords:** Smart House, Security, Privacy, Mobile Cloud Computing.

# Capítulo 1

# Introdução

Dispositivos inteligentes (*appliances)* são utensílios domésticos com poder de processamento, conectados via rede sem fio e, em geral, acessíveis via Internet [44]. Uma casa inteligente *(smart house)* possui *appliances* com o objetivo de prover diversos benefícios aos moradores, como comodidade, controlabilidade e eficiência energética [23]. Tais benefícios no cenário de *smart house* tornam-se mais relevantes quando combinados com *smart grid.*

*Smart grid* se refere ao serviço de distribuição de energia elétrica inteligente, estabelecido por um fluxo bidirecional de eletricidade e informações, capaz de administrar o fornecimento de energia [33]. O objetivo é a melhora do desempenho, confiabilidade e capacidade de interação, auxiliando a tomada de decisões de consumo por parte dos consumidores e do fornecimento de energia por parte da companhia de energia elétrica.

O sistema de *Supervisory Control and Data Acquisition* (SCADA) possibilita o controle e a aquisição de dados dos componentes de diversos serviços de infraestrutura [44]. Por exemplo, o SCADA permite viabilizar a adequação da carga de energia de uma rede elétrica, exigindo a reconfiguração do perfil (padrão) de consumo dos consumidores visando diminuir a chance de ocorrência de apagões energéticos.

Alguns trabalhos da literatura propõem acessos aos *appliances* diretamente da Internet, através de sistemas embarcados (*embedded system*) [1][14][37]. Os sistemas embarcados são utilizados neste cenário a fim de permitir que um morador altere o perfil de consumo dos *appliances* de forma remota [1][24][31]. Tais propostas permitem a reconfiguração (personalização) do perfil de consumo, mas impõem um custo adicional para manter os sistemas embarcados. Além disto, a exposição dos *appliances* na Internet pode

implicar em riscos à segurança e privacidade dos moradores. Os *appliances* podem ser atacados, explorando-se vulnerabilidades ou os dados dos moradores podem ser expostos (revelados) a terceiros usando a Internet [27].

Dispositivos móveis estão sendo usados para controlar remotamente os *appliances*, seja dentro ou fora de uma *smart house*, já que existem diversos aplicativos disponíveis para este fim [36]. Além disso, *smartphones* poderiam utilizar memória e processamento ociosos para formar uma *Mobile Cloud Computing* (MCC), modelo que integra a computação em nuvem com dispositivos móveis (*mobiles*). Pois, *mobiles* estão atualmente em uso por moradores da casa e permitem desempenhar as funcionalidades dos *sistemas embarcados* dedicados. Assim, permite-se diminuir os custos adicionais como a compra de novos equipamentos e otimiza-se o consumo energético, porque os mobiles já estão constantemente em uso.

## 1.1. Motivação

Com o gerenciamento de *smart devices* (*appliances)* e monitoramento energético, os moradores das residências podem controlar de maneira inteligente boa parte dos utensílios domésticos. Muitos trabalhos[1][2][14][15][18][23][47][48][51] visam fornecer essa comodidade, acessando esses *appliances* de diversas maneiras, inclusive de qualquer lugar via *web*. Desta forma, criam-se ambientes que expõem informações privativas e rotineiras dos moradores sem dar a devida atenção às soluções em segurança e privacidade dos moradores da *smart house*.

Os *appliances* da *smart house* não podem estar acessíveis diretamente a partir da Internet. Isso evita que um atacante os controle remotamente ou viole a privacidade dos moradores, monitorando seus hábitos de consumo para inferir que estão ausentes por exemplo.

Para garantir que pessoas mal intencionadas não acessem os *appliances* da *smart house*, é importante que exista dois critérios de segurança, (i) a pessoa que deseja acessar às informações deve estar nas proximidades da casa, e (ii) o acesso deve ser restrito a conexão local, sem acesso à Internet.

Este trabalho considera a hipótese de que é possível prover segurança e privacidade *by design* em *smart house*, gerido por uma integração de dispositivos móveis formando uma MCC.

## 1.2. Objetivo Geral

Este trabalho objetiva um gerenciamento energético em uma *smart house* tomando como critério de reconfiguração do perfil de consumo as informações recebidas pela *smart grid*. Tem também como intuito prover segurança e privacidade em uma *smart house*, isolando os *appliances* da Internet, mas sem deixá-los inacessíveis. A proposta provê economia financeira, porque evita a necessidade de aquisição de dispositivos embarcados para desempenhar as funções de controle da *smart house*.

## 1.3. Objetivos Específicos

a) Implementação da MCC em uma *smart house*;

b) Implementação de mecanismo de validação de confiança dos visitantes da *smart house*;

c) Garantir privacidade e segurança dos *appliances* isolando-os, mas mantendo-os acessíveis algumas funcionalidades a partir da Internet;

d) Criar perfis de consumo da *smart house*;

e) Viabilizar o consumo energético de maneira automatizada;

f) Implementar um protótipo, testar e avaliar alguns aspectos do esquema proposto.

## 1.4. Contribuições

O trabalho desenvolvido contribui para prover um esquema de gerenciamento seguro dos *appliances* de uma *smart house*. Através do método proposto, evita-se o gasto com a aquisição de novos equipamentos (dispositivos embarcados) e o desperdício energético para manter estes equipamentos ligados. Em seu lugar será utilizado o poder computacional ocioso dos *mobiles* dos moradores da casa.

O método criado gera um modelo seguro de admissão de novos participantes na MCC, considerando a confiança dos visitantes à admissão na MCC em relação aos moradores da casa. Considerando a mobilidade dos aparelhos utilizados, é possível realizar a coleta e processamento seguro dos *appliances* da casa através da MCC. A mobilidade é considerada, pois poderia inviabilizar o acesso a alguns *appliances* por estarem fora do alcance dos dispositivos embarcados que controlaria a casa, se a abordagem fosse diferente.

Na literatura, nenhuma abordagem com esquema de segurança e privacidade

desenvolvido *by design* controla uma *smart house* com exposição controlada dos *appliances* na Internet e utilizando uma MCC.

## 1.5. Estrutura do Documento

O capítulo 2 deste documento apresenta a fundamentação teórica necessária para a compreensão do restante do trabalho. O capítulo 3 contempla os trabalhos relacionados. O capítulo 4 apresenta a proposta deste trabalho e os resultados da sua avaliação. Por fim, no capítulo 5 há a conclusão deste estudo.

# Capítulo 2

# Fundamentação Teórica

Nesta seção serão apresentados uma introdução sobre *smart house*, os desafios relacionados à segurança e privacidade da *smart* house, a importância da *smart grid* e do *smart meter* no contexto da *smart house*. Além disso, será apresentado uma fundamentação em *cloud computing* e *mobile cloud* computing.

## 2.1 *Smart house*

Segundo Guinard e Trifa [25], objetos físicos irão interagir entre si através das redes de computadores. As interfaces irão proporcionar que utensílios enviem e/ou deixem disponíveis informações sobre ambiente, como por exemplo, sensores de movimento, temperatura, dentre outros. Desta maneira, objetos do cotidiano (e.g. máquinas de lavar, ar condicionado, geladeiras, televisões, etc) estarão conectados à Internet. A conexão destes objetos à rede mundial de computadores denomina-se Internet das coisas (*Internet of Things - IoT*) [4].

Com a tendência de que objetos do cotidiano ganhem "inteligência", o cenário da casa muda de forma drástica. O que inicialmente era feito de forma manual, torna-se possível através de um simples apertar de botão. A automatização de casas e a aplicação de sensores, que tem como objetivo transmitir informações do ambiente, caracteriza-se conceitualmente como *smart house* (Figura 2.1) [46]. Com a automação da casa, é necessário a existência de uma rede inteligente para gerenciar esses aplicativos [55].

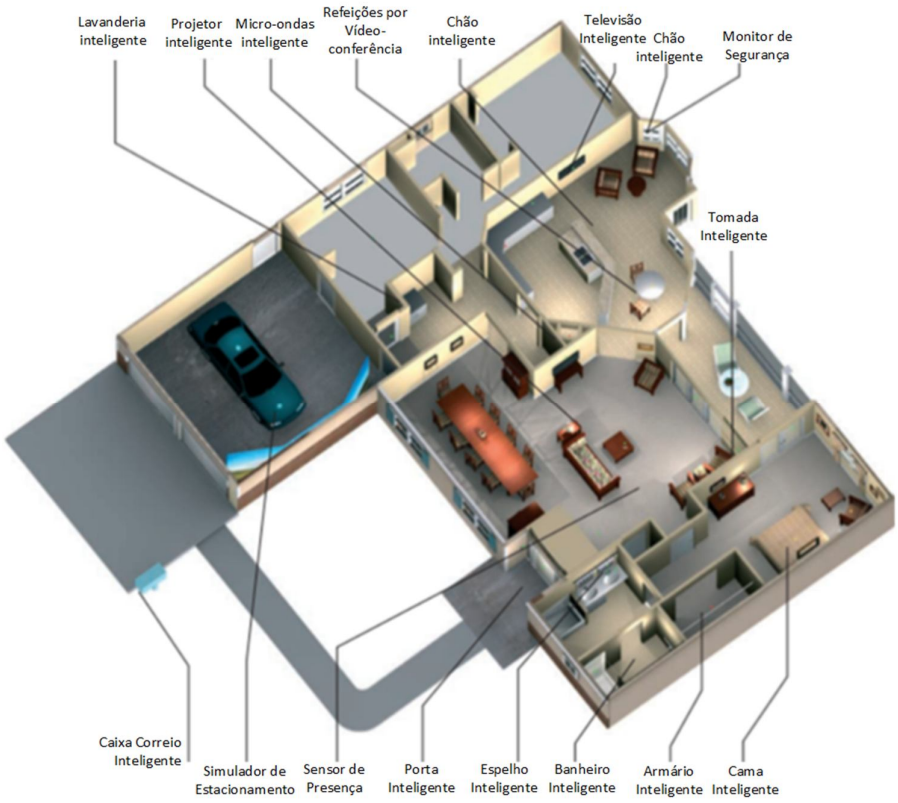


Figura 2.1- *Gator House.* Adaptado de [28].

Uma *smart house* possui um conjunto de dispositivos inteligentes denominados *appliances*, que também atuam como *things* na IoT [4]. Estes *appliances* interagem com os moradores e com o ambiente para otimizar seu funcionamento e melhorar a eficácia de suas funções [28]. Normalmente, os *appliances* possuem pouco recurso computacional e baixa capacidade de comunicação, de qualquer maneira as suas funcionalidades são providas remotamente, normalmente a partir de uma rede sem fio e serviços *web* [8]. Assim, um *appliance* é acessível pela Internet, possibilitando que um morador acesse os *appliances* ou que os próprios *appliances* acessem informações da Internet [44]. Por exemplo, um morador a caminho de sua *smart house* e seus *appliances* (e.g. ar-condicionado, cafeteira, conjunto de iluminação externa/interna) sejam acionados de forma remota e automática para melhor comodidade do morador.

O *appliance* de forma geral disponibiliza um conjunto variado de atributos, genéricos ou específicos, e podem ser obtidos ou atribuídos de acordo com a necessidade do usuário. Atributos genéricos são atributos comuns a todos os *appliances*, como o consumo instantâneo, por hora e diário. Já atributos específicos são dependentes do *appliance*, como o desgaste de

um componente ou atualização de um *firmware*. Embora a comunicação seja limitada, o tamanho e o número de requisições aos *appliances* pode ser significativo, variando de alguns atributos diariamente a centenas por segundo.

A *smart house* consiste em um sistema de vários *appliances* dispersos pela casa, interligados via cabo ou rede sem fio. Esses aparelhos podem ser classificados como [51]:

• Atuantes: Alarmes, luzes, portas, janelas, etc;

• Sensores: Calor, umidade, gás, movimento, etc;

• Atuantes/Sensores: Robôs, ar condicionado, máquina de lavar;

Um *appliance* possui diversos dados, como por exemplo, temperatura (graus), sensores de movimento (ativado ou não), etc. Esses dados devem estar seguros, garantindo assim a segurança e privacidade da casa e dos moradores [40].

## 2.2 Segurança e privacidade em *smart house*

Uma *smart house* possui *appliances* com o objetivo de prover diversos benefícios aos moradores, como comodidade, controlabilidade e eficiência energética [43]. Porém, essa comodidade traz desafios relacionados à privacidade, pois toda rotina, vida privada, atos e relações (Princípios da Privacidade) [59] - coletados dos *appliances* estarão registrados de alguma forma, seja em um servidor ou nos próprios *appliances*.

Os riscos à segurança e privacidade em *smart house* envolvem ameaças tanto da rede interna quanto da *smart grid (seção 2.3)*. Os *appliances* e *smart meter (seção 2.4*), podem ser operados de forma remota, tornando-os expostos a ameaças da Internet [40][30]. Um *appliance* pode ser desligado por um *smart meter* durante períodos de sobrecarga ou, por um aplicativo para *smartphone* acessado via Internet. Diversas operações podem ser realizadas ou dados podem ser acessados de um *appliance*, tornando uma *smart house* alvo de inúmeros ataques [27] (e.g. *rootkits*, *malwares* e *usage loggers*, dentre outros).

A privacidade é considerada uma questão crítica, uma vez que as informações armazenadas e trafegadas expõem os comportamentos e hábitos dos consumidores [40]. Determinadas atividades podem ser identificadas a partir de padrões de consumo elétrico, e.g. identificar se os moradores estão em casa, trabalhando ou viajando, assistindo televisão ou se possuem cerca elétrica [30].

A demanda por soluções para a segurança e privacidade torna-se cada vez mais evidente, não bastando utilizar soluções de criptografia, controle de acesso e autenticação,

mas sim uma perspectiva arquitetural voltada para a segurança e privacidade [19][30][33][40].

## 2.3 *Smart grid*

*Smart grid* envolve os aspectos relacionados a automatização da rede de energia elétrica, estabelecendo um fluxo bidirecional de fornecimento de eletricidade e informações a respeito da rede [33][62]. As informações da rede, como o estado de equipamentos (e.g. normal ou alarme) e o consumo individual dos consumidores podem ser obtidos em tempo real pela companhia [56]. A *smart grid* permite também que um consumidor altere seu perfil de consumo elétrico para se adequar a horo-sazionalidade tarifária, visando a diminuição de gastos em horários que a energia é mais cara.

De acordo com Li e Zhou [35] *smart grid* é um tema muito abrangente e envolve uma ampla gama de tecnologias como, medidor inteligente (*smart meter)*, geração distribuída, processamento de informação, *Power Line Communication* (PLC), entre outros. Ainda segundo esses autores *smart grid* é caracterizada como uma rede elétrica inteligente que agrupa estas tecnologias com o intuito de proporcionar uma rede elétrica com mais segurança, atendendo às necessidades financeiras e às demandas de energia do futuro.

A implantação de *smart grid* é uma evolução no sistema de distribuição de energia. Ela incorpora benefícios de computação e comunicação que permitem equilibrar de maneira quase que instantânea a oferta e demanda do sistema. É possível também o fornecimento do consumo em tempo real, reduzindo os índices de perdas, melhorando a qualidade da energia, eficiência energética e redução dos custos operacionais [34][56].

Uma *smart grid* é formada por milhares de dispositivos (inteligentes ou não) ligados por um sistema complexo de redes, e subdivididas em: *Home-Area Network* (HAN), rede interna de um consumidor; *Neighborhood-Area Network* (NAN), rede regional que engloba várias HAN; e *Wide-Area Network* (WAN), rede que engloba múltiplas NAN [19][33]. O alcance dos dispositivos e a taxa de transferência de dados são os principais fatores de classificação das redes, ilustrado na Figura 2.2.

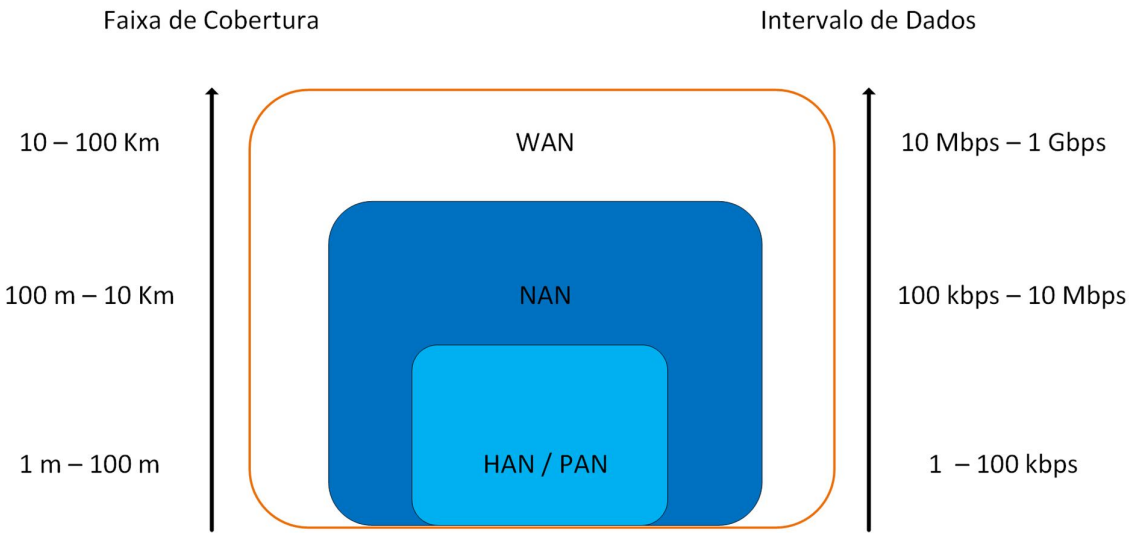


Figura 2.2 - Hierarquia de comunicações de redes inteligentes. Adaptado de [33].

O SCADA (*Supervisory Control and Data Acquisition*) é utilizado para efetuar o controle da rede de energia gerenciando a produção e a distribuição de energia de uma *smart grid*, controlando os dispositivos e possibilitando a interação com as *smart houses* [44][64].

## 2.4 *Smart Meter*

Segundo Depuru e colaboradores [16], *smart meter* é um medidor de energia que fornece informações do consumo energético de um consumidor em tempo real. Os *smart meters* são tidos como elementos-chaves para comunicação com as concessionárias de energia e estão diretamente relacionados à infraestrutura avançada de medição (*Advanced Meter Infrastructure* - AMI). O AMI é caracterizado como um elo para construção da *Smart Grid* [38].

O *smart meter* foi inicialmente aplicado aos grandes consumidores, como indústrias e grandes comércios, permitindo maior controle do consumo energético, estabelecendo uma granularidade fina na periodicidade das medições de consumo energético. Essas novas tecnologias ganharam destaque, e com isso os sistemas de medição eletrônica são empregados em diversos pontos da rede, como por exemplo, em pontos de intercâmbio de energia, sistemas de medição de subestações, unidades consumidoras e sistemas de média e alta tensão [22]. A utilização do *smart meter* para consumidores de pequeno porte levará a várias modificações nas relações de consumo[56].

Além de melhorar a precisão nas medições, o *smart meter* permite a comunicação instantânea com o consumidor. Desta forma, as redes inteligentes poderão se desenvolver e proporcionar interação entre o operador da rede e as cargas de cada unidade consumidora [64].

O *smart meter* pode ser empregado nas instalações elétricas de um consumidor, objetivando o monitoramento e controle de todos os *appliances*. Pode ser ainda utilizado como uma espécie de *gateway* de Internet, assim, a concessionária de energia pode ter controle remoto do consumo energético e acesso às informações, sem a necessidade de ir ao local. Os consumidores terão a possibilidade de saber em tempo real o valor de suas tarifas de energia, ajustando assim os seus hábitos de consumo [22].

De acordo com Depuru et al. [16], várias funcionalidades e vantagens ofertadas pelos medidores elétricos tornam atrativa a sua instalação. Por exemplo, informações mais claras e abundantes aos consumidores e distribuidoras, meios eficientes de combate à fraude, furto e inadimplência, perdas não técnicas, melhoras no processo de fiscalização, atuação remota, telemetria, novas modalidades de tarifação e qualidade da energia elétrica

Entretanto, qualquer mudança em larga escala pode levar a custos operacionais elevados, a dificuldade em reeducar os consumidores e os riscos de insucessos.

## 2.5 Computação em Nuvem

A computação em nuvem fornece capacidade de processamento, armazenamento, serviços e aplicações através da Internet, Figura 2.3. Além disso, a computação em nuvem permite a redução de custos e fornece flexibilidade em termos de provimento de recursos [57].

Na realidade, mesmo a computação em nuvem sendo um modelo computacional recente, a maioria das tecnologias utilizadas para a construção da computação em nuvem, como virtualização, *grid computing* e preços baseado na utilização – *pay-per-use* - não são novas. O que a computação em nuvem faz é utilizar-se desse conjunto de tecnologias existentes para executar tarefas de uma maneira diferente. Um provedor de serviços de infraestrutura em nuvem é tipicamente executado em um *datacenter* virtualizado onde há uma melhor utilização do *hardware.* A redução de servidores físicos no *datacenter* leva à redução dos custos de energia e refrigeração e também economia em *hardware* e de manutenção. A correta aplicação do modelo computacional em nuvem leva a otimização do *datacenter* dentro do campo empresarial [5].

Computação em nuvem pode ser definida como um grande conjunto de recursos virtualizados de fácil uso e acesso, tais como *hardware,* plataforma de implantação e/ou serviços [57].

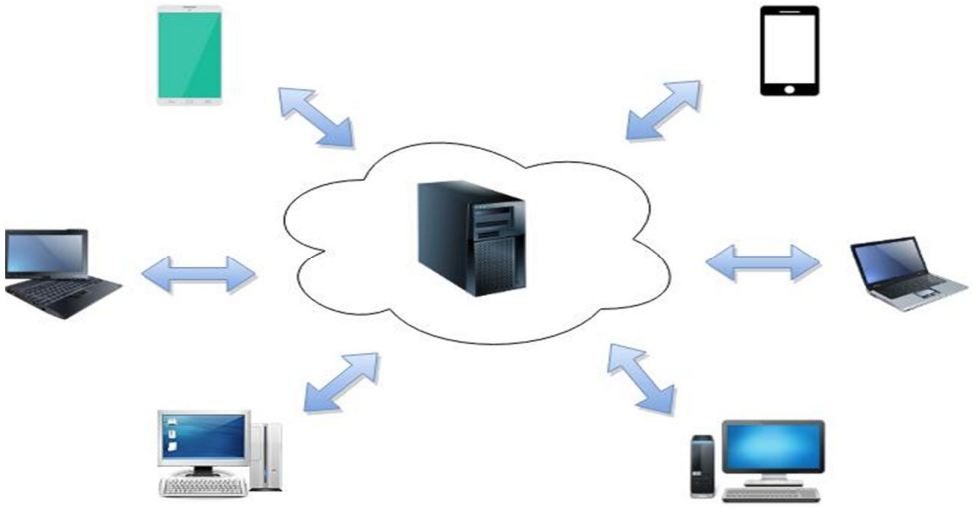

Figura 2.3 - A computação em nuvem com vários componentes. Adaptado de [52].

A correta aplicação do modelo computacional em nuvem leva a otimização do *datacenter* dentro do campo empresarial [5].

A computação em nuvem pode ser classificada quanto ao modelo de negócio, sendo classificado em Infraestrutura como Serviço (*Infrastructure as a Service* - IaaS), onde a nuvem oferece recursos computacionais de um data center de forma virtualizada, são exemplos de provedores de IaaS, *Amazon EC2*[1], *Datapipe*[2] e *Flexiscale*[3][57]. Pode ser classificado como *Software* como Serviço (*Software as a Service* - SaaS) e refere-se ao fornecimento de aplicações sob demanda através da Internet onde a aplicação é hospedada como um serviço ou aplicativo *web* e é disponível para usuários através de um navegador, como por exemplo, *Google Docs*[4] [65]. Também pode se classificar com Plataforma como Serviço (*Platform as a Service* - PaaS) onde oferece recursos para o consumidor implantar na infraestrutura da nuvem suas próprias aplicações, desde que utilizem linguagens de

[1] https://aws.amazon.com/pt/ec2/
[2] https://www.datapipe.com/
[3] http://www.flexiscale.com/
[4] https://www.google.com/docs/about/

programação e ferramentas suportadas pelo provedor, são exemplo de PaaS *Salesforce*[5] e *VisualForce*[6] [3].

A computação em nuvem também pode variar de acordo com o modelo de serviço que é aplicado. De acordo com Mell e Grance [42] pode ser classificado em Nuvem Privada, que é gerenciada pela própria empresa ou por agente externo, é muito utilizado em órgãos governamentais e bancos; Nuvem em Comunidade, onde a nuvem é compartilhada por diversas empresas; Nuvem Pública que é caracterizada pelo modelo *pay-per-use*, onde o usuário paga pelo serviço utilizado; e também a híbrida, que é a junção de dois ou mais modelos computacionais citados acima.

## 2.6 *Mobile Cloud Computing*

Dispositivos móveis que fornecem suporte a uma ampla gama de aplicações como jogos, processamento de imagens, processamento de vídeo, *e-commerce* e serviços de rede social *online* também estão ganhando enorme popularidade [29]. De qualquer maneira, os avanços de *hardware* nos *mobiles* e a vida útil da bateria não tem acompanhado a crescente exigência computacional da evolução das aplicações ao longo dos anos. Desta forma, muitas aplicações ainda permanecem inadequadas para *mobiles* devido a restrições da plataforma como o baixo poder de processamento, memória limitada, conectividade de rede imprevisível, e vida útil da bateria limitada [32].

De acordo com Satyanarayanan e colaboradores [53] o principal objetivo da *Mobile Cloud Computing* (MCC) é usar a capacidade de processamento e armazenamento da computação em nuvem tradicional para aliviar o gasto energético e tempo de processamento de aplicações que demandariam mais esforços para serem realizadas. Desta forma, as restrições dos *mobiles* podem ser abordadas, e tarefas que demandam grande capacidade de processamento podem ser terceirizadas para sistemas mais capazes. [17].

Assim, segundo Khan e colaboradores [29], a MCC é definida como uma integração de computação em nuvem tradicional com dispositivos móveis, tornando-os suficientes em questões de poder de processamento, memória e energia. Além disso, a computação em nuvem pode permitir novos serviços móveis inteligentes, usando as informações de contexto coletadas a partir de sensores de dispositivos para fornecer serviços personalizados [59]. Um

[5] http://www.salesforce.com/br/
[6] https://developer.salesforce.com/

exemplo seria o aplicativo *Waze*[7], que utiliza sensores dos dispositivos móveis para monitoramento de trânsito.

De acordo com Fernando e colaboradores [20] a MCC pode ser dividida em 3 diferentes conceitos. Na primeira, ilustrado na Figura 2.4, os *mobiles* agem como clientes e as aplicações são executadas em um servidor em nuvem, como por exemplo o *Facebook*[8], *Twitter*[9] e *Google Translate*[10].

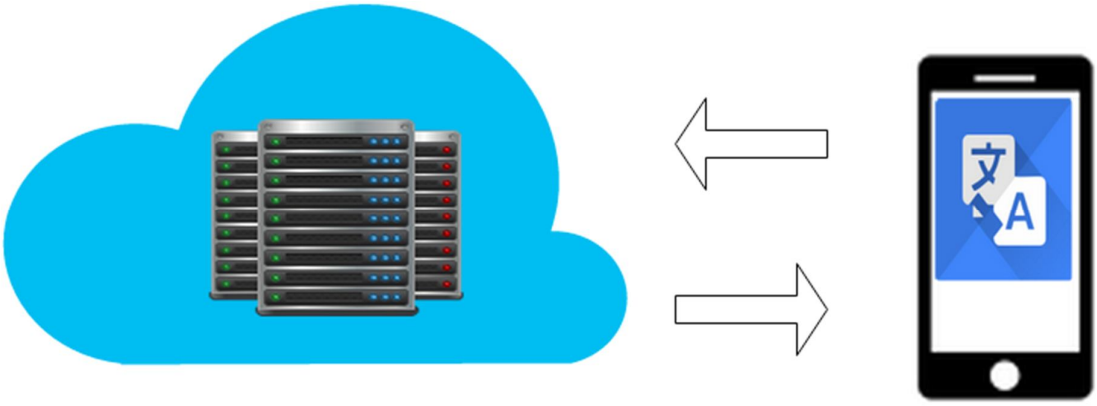

Figura 2.4 - *Mobile* utilizando recursos de um servidor em nuvem. Adaptado de [20].

Em uma segunda abordagem, Figura 2.5, o *mobile* é utilizado como um servidor da MCC, através de conexões *peer-to-peer* [39].

[7] https://www.waze.com/pt-BR
[8] https://www.facebook.com
[9] https://twitter.com/
[10] https://translate.google.com.br

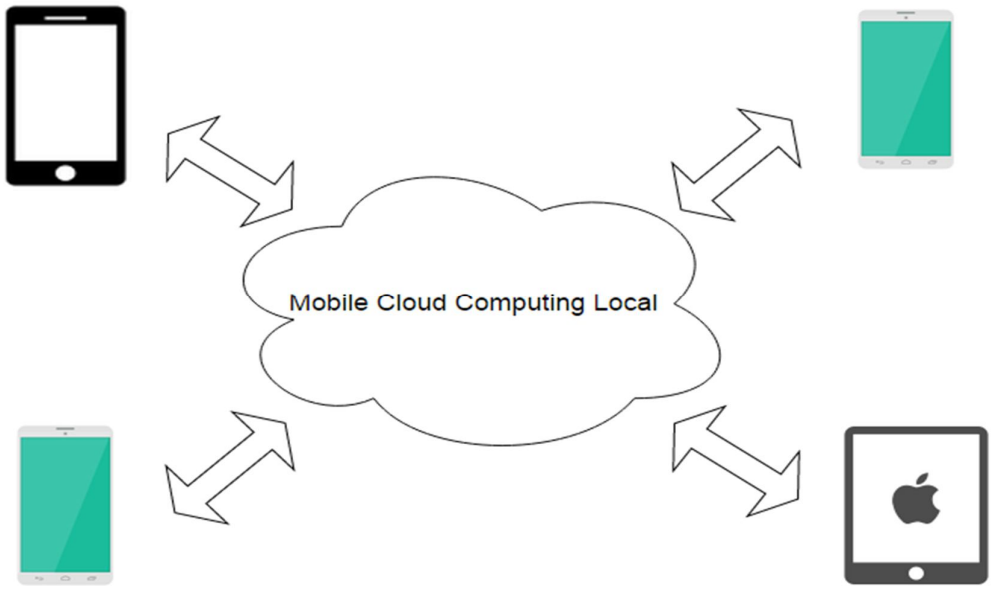


Figura 2.5 - Nuvem formada por dispositivos próximos. Adaptado de [20].

A terceira é um modelo proposto por Satyanarayanan e colegas [53], Figura 2.6, que utiliza *cloudlets*, servidor local que processa ou encaminha as informações para servidores na nuvem.

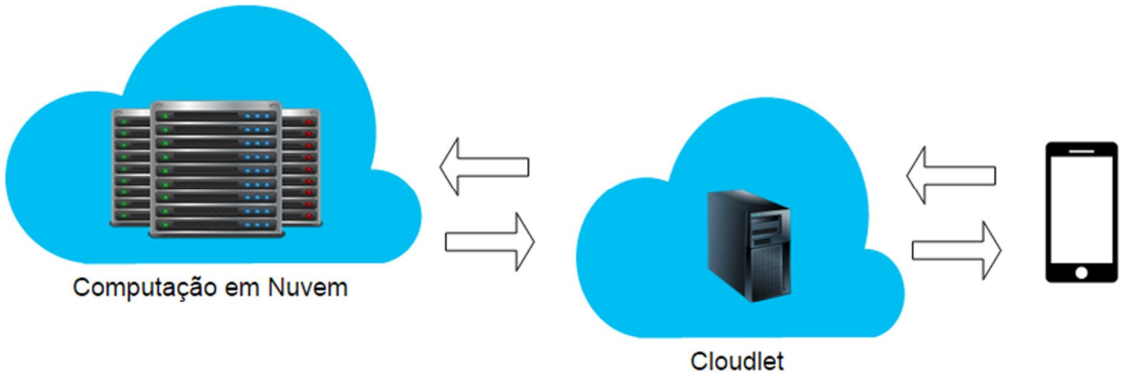


Figura 2.6 - *Cloudlet*. Adaptado de[20].

Apesar dos obstáculos que os sistemas de computação móvel inevitavelmente enfrentam em relação a sistemas estacionários, incluindo as limitações de recursos, o risco de perdas e danos, a variabilidade em termos de conectividade e energia finita, Marinelli [39] expõe numerosas vantagens da computação em nuvem em um *hardware* móvel, sendo elas:

- Dados móveis como sensores de *logs* e dados multimídia estão imediatamente disponíveis e podem ser processados no local ou em outro nó que está próximo. Isso elimina a necessidade de transferir dados para serviços centralizados.

- Os dados podem ser partilhados mais rapidamente entre dispositivos móveis através da área local ou redes *peer-to-peer* (P2P). A distribuição dos dados através da rede local evita o envio dos dados a servidores terceiros e suscetíveis a exposição de dados privados
- Serviços que utilizam dados móveis, podem ser criados com pouca infraestrutura nos servidores. A carga computacional pode ser distribuída entre os dispositivos móveis, não exaurindo assim o servidor do serviço provido.
- A crescente quantidade de dispositivos móveis em uso facilita a escalabilidade da nuvem, diferentemente da abordagem em nuvem tradicional.
- A propriedade do *hardware* é distribuída. Usando *hardware* móvel de propriedade de muitas pessoas diferentes, os riscos que surgem quando os serviços em nuvem de propriedade são utilizados, tais como dados de *lock-out* e dependência de entidades externas para a privacidade de dados são evitados.

# Capítulo 3

# Trabalhos Relacionados

Os trabalhos relacionados foram divididos em 6 subseções: (i) apresenta o controle da smart house através de dispositivos locais; (ii) aborda os trabalhos que proporcionam o acesso aos appliances da smart house através da Internet; (iii) disponibiliza os trabalhos que visam segurança e privacidade a smart house; (iv) relaciona os trabalhos envolvidos à redução de consumo energético; (v) aborda a MCC e (vi) concluí os trabalhos relacionados**.**

## 3.1 Acesso a *smart house* via Internet

Diversos trabalhos abordam o controle dos *appliances* de uma *smart house*. Alguns autores [2][28] apresentam uma *smart house* que possui um servidor dedicado responsável por controlar e monitorar os *appliances*. Alguns trabalhos [1][2][18][24][37][51] gerenciam os *appliances* utilizando rede sem fio e um computador atuando como servidor na *smart house*. Assim, os moradores podem controlar seus *appliances* localmente e via Internet. Porém, esse acesso expõe a *smart house* a possíveis ataques, possibilitando o acesso não autorizado aos *appliances*.

Da mesma forma, Perera e colegas [47], propuseram um *middleware*, denominado MOSDEN, capaz de coletar e processar os dados dos sensores através dos dispositivos móveis. Os autores descrevem que seria impraticável acessar os *appliances* diretamente da Internet devido ao impacto negativo em termos de computação, uso de banda e custo de *hardware*. Assim, utilizam o *middleware* para o envio dos dados ao servidor, seja ele na nuvem ou local. Utilizar um servidor para o gerenciamento dos *appliances* implica em um ponto adicional de consumo energético e, na necessidade de hospedagem de um servidor *web*.

No trabalho de Perera e colaboradores [48] é proposto a utilização de *smartphones* para a coleta e processamento de dados dos *appliances*, que posteriormente são armazenados em um servidor local ou em nuvem. Permitir que os *appliances* sejam acessíveis via Internet, seja diretamente ou via *smartphone,* expõe a *smart house* além de adicionar um ponto único de falha, seja localmente ou em nuvem. Da mesma forma [26][44][60] utilizam um sistema embarcado e servidores para gerir os *appliances*, culminando em ponto único de falha e custo adicional para aplicar esse sistema.

Por outro lado Korkmaz e colegas [31] propõe uma arquitetura diferente, o autor retira o servidor do ambiente físico da *smart house* e hospeda na nuvem, comunicando com os *appliances*, e ainda com a possibilidade de atender mais de uma *smart house*. Da mesma forma, o acesso oriundo da Internet expõe a *smart house* aos ataques, possibilitando o acesso não autorizado aos *appliances* e como o servidor é hospedado na nuvem aumenta os gastos da implantação da *smart house.*

## 3.2 Acesso local à *smart house*

Outros trabalhos [10][50][62] sugerem que o acesso aos *appliances* ocorra via *bluetooth*. Utilizar uma comunicação de baixo alcance, como *bluetooth*, delimita a distância de comunicação com os *appliances*, isto implica em uma vantagem a nível de segurança e baixo custo de implantação. Esses trabalhos utilizam uma conexão direta entre dispositivos portáteis com sistema operacional Android e os *appliances*, isso pode implicar em uma perda de comunicação devido ao baixo alcance do *bluetooth*. Já em [14][54] a comunicação entre os *appliances* é intermediada por um controlador, implicando em um ponto único de falha.

Perera e colaboradores [48] desenvolveram um aplicativo para *smartphone* chamado de *SmartLink*, onde o *smartphone* é capaz de descobrir e configurar os *appliances*. A solução necessita um dispositivo (IoT) intermediário que realiza a comunicação direta com os *appliances*. De acordo com autores o modelo proposto precisa que os três componentes (*appliances*, *smartphones*, e *middleware*) funcionem coletivamente para o sucesso da solução. Desta forma, a solução precisa de vários fatores para que a aplicação opere corretamente, ocasionando em diversos pontos de falha.

## 3.3 Segurança e Privacidade em *smart house*

Uma *smart house* traz desafios relacionados à privacidade das informações. As informações coletadas dos *appliances* ficam armazenadas em sua memória ou até mesmo em um servidor, como por exemplo dados de vida privada, hábitos, ações e as relações. De acordo com Warren e Brandeis [59] esses dados se referem aos princípios de privacidade e devem ser considerados.

O acesso não autorizado às informações coletadas pelos *appliances* implica em uma grave falha de segurança. De acordo com Cook [11], moradores hesitam em introduzir sistemas de detecção e monitoramento em suas residências, por medo que pessoas não autorizadas acessem e utilizem o ambiente de maneira inapropriada. De maneira semelhante, Demiris et al. [15] também observam as preocupações dos moradores em relação a privacidade, argumentando que as tecnologias que detectam e monitoram as atividades dentro da casa podem ser vistas como intrusivas no ambiente residencial.

O trabalho de Witkovski e colaboradores [61] apresenta um método de autenticação baseado em chave e em gerenciamento de identidade (*Identity Management* - IdM) para prover *Single Sign-On* (SSO) em IoT. Embora o trabalho permita o acesso seguro aos *appliances* para manipular atributos específicos, o problema de gerenciamento de energia e privacidade para a *smart house* no contexto de *smart grid* não é tratado.

O trabalho de Chakravorty [7] trata a questão de privacidade da *smart house*, apresentando um sistema de coleta segura e armazenamento dos dados dos *appliances* em um *cluster* distribuído utilizando o *framework Hadoop*[11]. Apesar da solução levar em conta fatores de segurança e privacidade para que o morador acesse suas informações através da Internet, essa solução torna-se inviável na prática, visto que necessita de unidades de processamento, armazenamento e controles de acesso. A solução não considera a coleta dos dados dos *appliances*, dessa maneira os dados podem não chegar confiáveis ao primeiro ponto da sua estrutura.

## 3.4 Redução do consumo energético da *smart house*

O trabalho de Han e colegas [26] propõe uma arquitetura para redução de energia que controla as tomadas (*smart outlet)* da *smart house*. Esse projeto desliga os aparelhos que estão

[11] https://hadoop.apache.org/

em *standby* e controla a intensidade de luminosidade nos cômodos, evitando o consumo de energia desnecessário.

O trabalho de Mrazovac e colaboradores [44] propõe um controlador que pode ser acessado localmente ou via Internet, que desliga equipamentos em *standby* a partir de padrões de uso para utilização de energia.

Já o trabalho de Weiss e Guinard [60] monitora todo o consumo energético da *smart house* momentaneamente através de um *gateway* que concentra todas as informações das tomadas inteligentes e torna esses *appliances* disponíveis para *smartphones* e computadores, a fim de que o morador gerencie o consumo elétrico da casa através da Internet.

## 3.5 *Mobile Cloud Computing* (MCC)

Motivado a reaproveitar recursos disponíveis nos *mobiles*, Kristensen [13] propôs um sistema que particiona tarefas entre dispositivos móveis (*smartphones*, *tablets* e *notebooks*) formando uma MCC local. Nesse mesmo desafio, para garantir que a tarefa seja realizada com sucesso, independente se o dispositivo tenha recursos necessários, Satyanarayanan e colaboradores [53], propõe uma *cloudlet*, técnica baseada em aumento de desempenho, na qual uma máquina virtual alocada em um servidor local permite que dispositivos móveis operem como cliente executando as tarefas.

Cuervo e Balasubramanian [12] propuseram um *framework* que realiza o cálculo do custo benefício de processar a aplicação em um servidor em nuvem, levando em consideração o tipo de conexão (3G ou rede sem fio) e características do processamento da aplicação, visando o menor gasto energético, porém não descartando as prioridades dos usuários.

Outros trabalhos propõem utilizar *mobiles* para a realização de tarefas de forma distribuída e compartilhada, abordagem considerada como uma MCC no contexto de *crowdsourcing* [6][9]. Alguns *frameworks* são propostos no contexto de MCC local, porém não são disponibilizados ou utilizam uma arquitetura que depende de um servidor estático [6] [9][41][63].

## 3.6 Considerações

Embora várias propostas visem os benefícios da *smart house*, principalmente permitindo o gerenciamento dos *appliances*, as abordagens citadas apresentam fragilidades quanto a segurança e privacidade. Além disso, as propostas não tratam a necessidade de

interação com a *smart grid*. Não considerando a possibilidade de uma reconfiguração de consumo requisitada pela companhia elétrica, ou a mudança de perfis de consumo de acordo com modelos tarifários, visando a diminuição de gastos em horários que a energia é mais cara.

A tabela a seguir apresenta os trabalhos relacionados e apresenta suas respectivas abordagens em relação ao trabalho proposto.

Tabela 3.1 - Abordagens dos trabalhos relacionados

| | **Gerencia *Smart House Localmente*** | **Gerencia *Smart House Externamente*** | **Controle Energético** | **Mecanismo Segurança aos *Appliances*** | **Sistemas Embarcados** | **MCC** |
|---|---|---|---|---|---|---|
| Helal et al. [28] | Sim | Sim | Não | Não | Sim | Não |
| Alkar e Buhur. [2] | Sim | Sim | Não | Sim | Sim | Não |
| Elshafee e Hamed [18] | Sim | Sim | Sim | Sim | Sim | Não |
| Liang, et al. [37] | Sim | Sim | Sim | Não | Sim | Não |
| Rajabzadeh, et al. [51] | Sim | Sim | Não | Sim | Sim | Não |
| Golzar e Tajozzakerin [24] | Sim | Sim | Sim | Sim | Sim | Não |
| Al-Ali, e AL-Rousan [1] | Sim | Sim | Não | Sim | Sim | Não |
| Perera et al. [48] | Sim | Sim | Não | Não | Não | Não |
| Han, et al. [26] | Sim | Não | Sim | Não | Sim | Não |
| Mrazovac et al. [44] | Sim | Sim | Sim | Não | Sim | Não |
| Weiss e Guinard [60] | Sim | Sim | Sim | Não | Sim | Não |
| Korkmaz et al. [31] | Sim | Sim | Sim | Sim | Sim | Não |
| **Proposta deste trabalho** | **Sim** | **Não** | **Sim** | **Sim** | **Não** | **Sim** |

Fonte: Autoria própria.

# Capítulo 4

# Proposta

Este trabalho propõe a criação de um esquema de segurança, baseado em *mobile cloud computing* (MCC) que oferece privacidade *by design*, num cenário de gestão do consumo energético da *smart house* para atender as necessidades do morador e da companhia de energia elétrica.

De maneira periódica, a MCC coleta e processa informações de consumo dos diversos *appliances* disponíveis na *smart house* e consolida os respectivos perfis de consumo. Assim, gera-se um mecanismo descentralizado de processamento e coleta de dados que usa os recursos de processamento dos moradores, sem a necessidade de um elemento exclusivo, como um servidor dedicado (sistema embarcado) para a gestão da *smart house*. Atualmente, os *mobiles* tem assumido o papel de controle remoto de muitos *appliances*, devido à facilidade de instalação de aplicativos oferecidos pelos fabricantes e meios de comunicação sem fio que eles possuem.

A utilização da MCC implica em várias vantagens em nível de segurança. Primeiramente esta proposta não possui um ponto único de falha, considerando que a MCC é composta de vários *mobiles* e a função de coordenação da MCC não é atribuída sempre ao mesmo *mobile*, desta forma, inibe-se um ataque prévio a um possível coordenador. Além disso, os *mobiles* possuem aplicativos e configurações de segurança próprias, pelos quais o morador pode optar. A comunicação entre os *mobiles* é considerada de curto alcance, isso garante que apenas os *mobiles* que estejam fisicamente nos arredores da casa tenham possibilidade de acessar a MCC. Além disto, assumimos que os serviços que executam em periféricos do *smart meter*, não tem alcance para acessar os *appliances* que estão espalhados pela casa toda. Desta maneira, o comprometimento do *smart meter* não compromete os *appliances* da *smart house.*

Na proposta (Figura 4.1) a MCC é composta por três tipos de participantes, (i) *Resident*: *mobile* de um morador da casa - realiza a coleta e processamento das informações de consumo dos *appliances*; (ii) *Header: mobile* coordenador da MCC - responsável pelo armazenamento das informações de consumo e sua consolidação nos perfis de consumo e (iii) *Visitor: mobile* de um visitante confiável dos moradores da casa - utilizado apenas para coletar as informações para a MCC. Obrigatoriamente a MCC deve possuir um *Header*, a seção 4.2 aborda como se realizada a sua escolha, assim como a definição de visitante confiável.

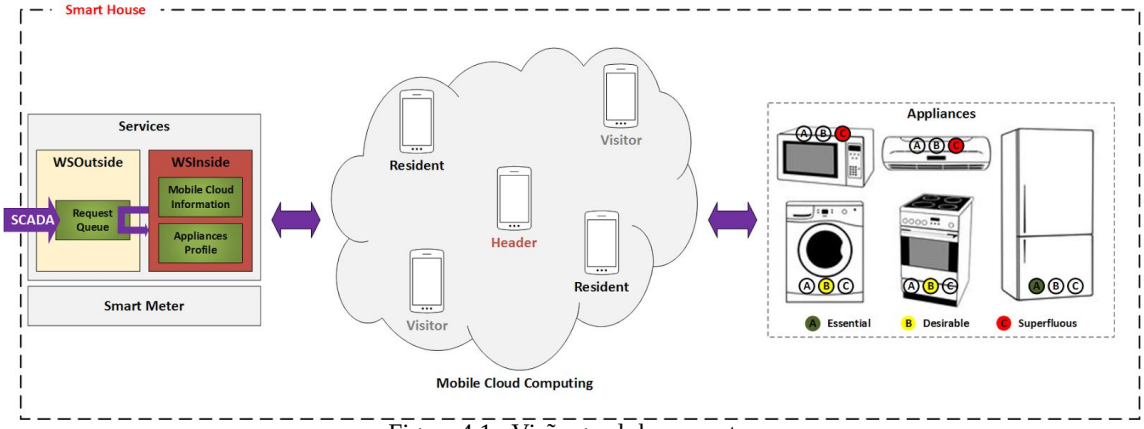


Figura 4.1– Visão geral da proposta.

A MCC atua como elemento central da arquitetura (Figura 4.1), responsável por coletar as informações de consumo dos *appliances* de maneira periódica. Informações são os diversos atributos dos *appliances*, podendo ser genéricos ou específicos. Atributos genéricos são atributos comuns a todos os *appliances*, como o consumo instantâneo, por hora e diário. Já atributos específicos são dependentes do *appliance* e foram considerados no trabalho de Witkovski e colegas [61].

A coleta de dados ocorre de forma distribuída, coordenada pelo *Header*. A partir das informações obtidas de cada *appliance*, a MCC executa seu processamento de maneira distribuída para consolidar o consumo da *smart house*. Os *mobiles* do tipo *Visitor* estão limitados a coleta e transmissão de dados dos *appliances* para evitar a violação de privacidade da casa.

O processamento dos dados da casa fornece informações consolidadas para o controle do consumo de energia da *smart house*, como por exemplo, o consumo atual dos *appliances*. Com essas informações torna-se possível gerar padrões de utilização dos *appliances* ao longo

do dia, semana, mês e ano, afim de mapear perfis de consumo de energia elétrica. Os perfis de consumo englobam informações relativas aos *appliances* da *smart house*, como consumo, horário de maior utilização, entre outros. Essas informações aliadas à necessidade de cada morador permitem classificar os *appliances* em três possíveis categorias: essenciais, supérfluos e desejáveis.

A classificação automática de *appliances* serve como uma sugestão para o morador, podendo ser definida a critério do mesmo. A classificação será utilizada no processo de redução de consumo de energia, discutido no capitulo 5.

Os perfis de consumo são armazenados pelo *Header* no *WSInside*. O *WSInside* é um serviço disponível apenas dentro da *smart house*, responsável pelo armazenamento dos perfis de consumo, controle da MCC e garantia da sua singularidade (existência única numa *smart house*). Em caso da ausência da MCC, atende os pedidos de reconfiguração de consumo (feito com base nos perfis armazenados). Já o *WSOutside* intermedia a comunicação entre o SCADA e o *WSInside*, recebendo as solicitações de redução de consumo.

A arquitetura ainda conta com a utilização de dois *web services* para evitar que ataques oriundos da Internet comprometam a segurança e privacidade da *smart house*. O SCADA só pode depositar requisições no *WSOutside. Assim,* o *WSInside* periodicamente, lê os pedidos de uma fila de requisições. Dessa maneira, a *smart house* é isolada da Internet porque não há acesso interativo a *smart house* a partir da Internet.

## 4.1 Utilização da *Mobile Cloud Computing*

Na instalação da arquitetura, realizada pela companhia de energia, o técnico da companhia cadastra um morador no sistema para administrar os demais moradores da *smart house*. No cadastro, são definidas informações referentes ao *mobile* do morador, além da definição de uma senha de acesso a MCC. Dessa forma, os moradores tornam-se *Resident* para a MCC.

Assim que um morador que porte um *mobile* do tipo *Resident* alcançar o perímetro de cobertura da rede sem fio da casa, o aplicativo iniciará um processo para ingresso na MCC (Figura 4.2). Primeiramente, o *Resident* envia uma solicitação de autenticação para o *WSInside,* informando suas credenciais que foram definidas no cadastro (evento 1). O *WSInside* verifica as credenciais e retorna o status da autenticação, caso seja válida o *Resident* buscará o *Header* da MCC (evento 2). O *WSInside* informa os dados do *Header,* caso exista

um. Se não existe *Header*, o *Resident* solicitará ao *WSInside* que inicie o processo de eleição de *Header* (evento 3).

O processo de eleição de *Header* pode ocorrer em dois casos, quando uma MCC está sendo criada ou quando o *Header* de uma MCC fica indisponível. Um *Header* pode ficar indisponível por vários motivos, visto que uma das características de um *mobile* é a sua mobilidade. Dessa forma, é necessária uma política para a escolha de um novo *Header*. Para simplificar, utilizou-se uma política de escolha aleatória, ou seja, qualquer participante do tipo *Resident* pode se tornar um coordenador. A escolha aleatória traz como principal benefício a impossibilidade de prever o próximo *Header*, avaliada na *seção 6.2*. Dessa forma, um possível atacante de sistema não poderá direcionar o ataque a um único *Resident*, visando o controle da MCC. Entretanto, a arquitetura proposta permite adotar uma política de eleição mais robusta, como as tradicionais políticas de sistemas distribuídos ou baseadas em recursos de *hardware*, onde o *mobile* com melhor *hardware* é o *Header*. É importante ressaltar que apenas os *Residents* podem se tornar *Header*. As informações relativas ao *Header* da MCC são informadas apenas a um *Resident*, que envia uma mensagem de registro para entrar na MCC (evento 4).

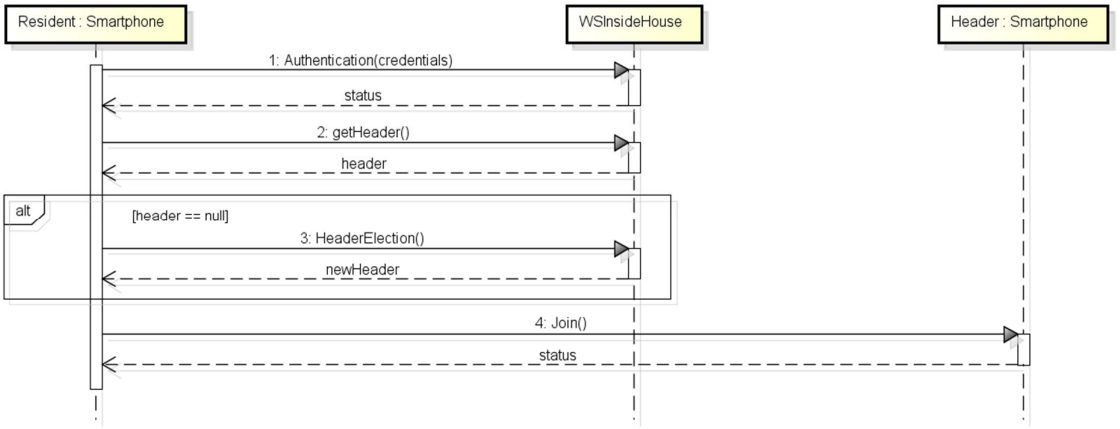


Figura 4.2 – Diagrama de sequência para Resident ingressar na MCC

Quando um visitante deseja entrar na MCC, o processo é diferente (Figura 4.3). Primeiramente, o *Visitor* solicita participar da MCC para o *WSInside* (evento 1). Caso exista uma MCC, o *WSInside* obtém o *Header* e solicita a participação do *Visitor* (evento 1.2). O *Header* precisa avaliar o nível de confiabilidade do *Visitor* e para isso apresenta ao morador, como sugestão, o nível de confiança do *Visitor* (evento 1.2.1). O nível de confiança pode ser

calculado através do histórico de mensagens e ligações que algum *Resident* teve com o *Visitor*, em um determinado período de tempo. Caso o *Visitor* seja considerado confiável, atingindo um *threshold*, o *Header* manda um SMS (evento 1.2.2) para o *Visitor* com um *nonce* (sequência de números aleatória). O visitante informa ao aplicativo o *nonce* acrescido para que seja enviado ao *Header*. Esse processo garante o nível de confiança do *Visitor*, confirmando que o número de telefone do *Visitor* e o do celular presente na casa, utilizado para determinar se é confiável, seja o mesmo. Com a confirmação da identificação do *Visitor*, o *Header* registra o *Visitor* na MCC (evento 1.2.3).

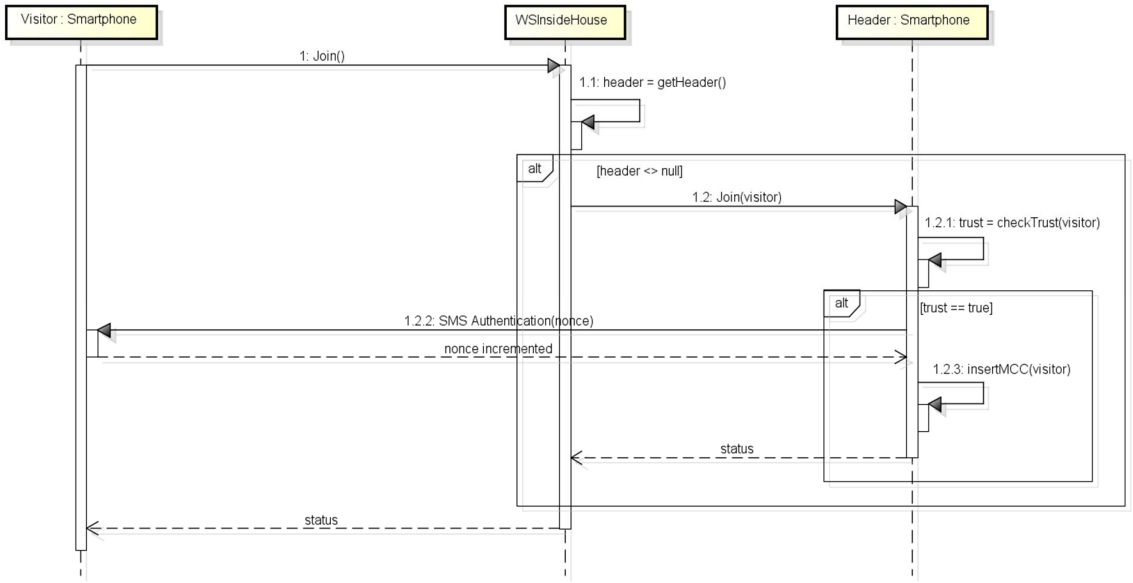


Figura 4.3 - Diagrama de sequência para *Visitor* ingressar na MCC.

Para efeitos de otimização, após o *Visitor* entrar na MCC, sua participação tem validade por um tempo determinado, por exemplo, 24 horas. Caso o *Visitor* saia da MCC e volte após esse período, será necessário realizar a validação de confiança novamente.

# Capítulo 5

# Cenário

A proposta deste trabalho assume que a companhia de energia elétrica possui um sistema (SCADA) integrado a *smart house*, podendo fazer coletas consolidadas de consumo de toda uma região e somá-las a outras regiões, de modo a obter o consumo total de uma localidade em dado instante, sendo então capaz de prever um apagão *(blackout*). Pois, quando o consumo cresce em ritmos críticos, o SCADA irá solicitar que os consumidores reduzam parcialmente o consumo energético para reduzir a carga do sistema e evitar o possível *blackout*, por exemplo.

A Figura 5.1 representa o cenário considerado nessa proposta, onde o SCADA encontra-se conectado com diversos *Gateways*. Um *Gateway* faz o intermédio entre o SCADA e diversas *smart houses.* O *Gateway* tem como objetivo consolidar as informações de consumo de energia elétrica de determinada região. Assim, o SCADA não conversa diretamente com as *smart houses,* aumentando a granularidade das informações coletadas e a privacidade dos moradores. O *Gateway*, por sua vez, está conectado as *smart houses* através do *WSOutside*.

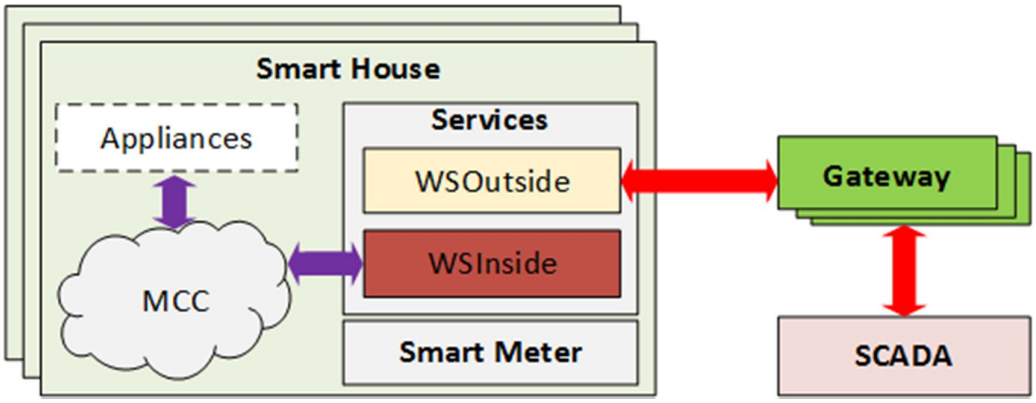


Figura 5.1- Visão geral da integração da *smart house* à *smartgrid*

Após o SCADA prever que a energia elétrica disponível não é suficiente para atender toda a demanda necessária, o SCADA inicia o processo de solicitação de reconfiguração do consumo de energia elétrica. O processo consiste, primeiramente, em determinar as regiões que serão alvos da redução e obter o endereço dos *gateways* vinculados. Baseado em uma equação estatística, o SCADA determina e informa a porcentagem de redução de consumo que cada *gateway* deve alcançar. Assim, o *gateway* inicia um processo de solicitação de redução do consumo energia com cada *smart house*, usando o *WSOutside*.

Dessa forma, o *gateway* solicita a redução de consumo de energia ao *WSOutside* de cada *smart house*, informando o valor de redução e o tempo limite para isso ocorrer. Essa solicitação ficará armazenada no *WSOutside*. O *WSInside* consulta periodicamente o *WSOutside*, em busca de solicitações da companhia de energia. Quando a consulta do *WSInside* retorna uma solicitação da companhia de energia, o *WSInside* verifica se existe MCC na *smart house*. Caso exista a MCC, o processo transcorre normalmente. A MCC lê a fila de requisições do *WSInside*, trazendo como benefício a segurança e a proteção de privacidade, pois se houver alguma vulnerabilidade, não poderá ser explorado porque o processamento não é interativo e portanto não acontecerá a exposição de dados.

A partir da requisição de redução consumo, a MCC analisa, baseada nas classificações dos perfis, quais *appliances* precisam ser reprogramados ou até mesmo desligados. Alguns *appliances* tem a possibilidade de reduzir sua potência, assim, economizam energia. Dessa forma, primeiramente a MCC prioriza a redução/desligamento dos *appliances* considerados supérfluo, depois os desejáveis. Caso não exista a MCC na *smart house*, o perfil de consumo memorizado no *WSInside* será utilizado para reduzir o consumo.

Observe que neste caso não ocorre interação dos mobiles com os *appliances*. O que acontece é o desligamento de cada um dos circuitos de alimentação dos *appliances* de cada perfil de consumo, iniciando com o supérfluo e se for necessário desejável.

Quando o tempo definido pelo *gateway* expirar, o *gateway* verifica se as solicitações de redução de consumo foram atendidas, solicitando a coleta de consumo instantâneo da *smart house*. Caso a requisição não tenha sido atendida, o *gateway* atua da forma tradicional, cortando a energia da *smart house*.

Além disso, o *gateway* possui uma *black list* que contém as *smart houses* que não atenderam a solicitação de redução de consumo - essas *smart houses* serão as primeiras a terem a energia desligada, beneficiando as *smart houses* que tem colaborado com os pedidos de redução disparados pelo SCADA.

# Capítulo 6

# Protótipo

A implementação do protótipo é baseada no *web service WSInside* e na MCC. O *WSInside* foi implementado em java utilizando a biblioteca JAX-RS[12]. A MCC foi implementada através da integração de aplicativos desenvolvidos em *mobiles* baseados em Android[13]. Para reproduzir o comportamento dos *appliances*, desenvolvemos uma aplicação em java que responde as requisições de consumo realizadas pelos *mobiles* da MCC.

Para o desenvolvimento dos aplicativos que compõem a MCC foi utilizado Jelly Beans (v4.1) do Android, requerida pelo NSDiscovery[14] - mecanismo de anúncio em *broadcast* e descoberta de serviços na LAN. Participantes da MCC anunciam serviços de coleta e processamento das informações obtidas pelos *appliances* usando NSDiscovery. O *Header* não anuncia o serviço, apenas descobre os mobiles que desejam ingressar na MCC. Dessa forma, não é possível descobrir quem é o *Header*. Apenas os *mobiles* que já estão ativos na MCC tem o conhecimento de quem é o *Header* (evento 2, da Figura 4.2). Os participantes da MCC (*Resident*) são cadastrados no *WSInside*, onde são armazenados o número telefônico, IMEI (*International Mobile Equipment Identification*), MAC *address* e uma senha definida por cada um. Essas informações são validadas a cada ingresso do *Resident* na MCC.

Quando um *Visitor* solicita ao *WSInside* o ingresso na MCC, deverá informar o seu número de telefone. Então, o *WSInside* encaminha o pedido ao *Header*, se o *Visitor* é considerado confiável através do processo de avaliação de confiança, o *Header*, lhe envia uma

[12] https://jax-rsspec.java.net
[13] https://www.android.com
[14] http://developer.android.com/training/connect-devices-wirelessly/nsd.html

mensagem SMS (*Short Message Service*) de validação (evento 1.2.2 da Figura 3), utilizando a biblioteca *telephony*[15] do Android.

Cada aplicativo instalado no *mobile* possui um banco de dados SQLite[16], responsável por armazenar os dados coletados dos *appliances* que estão ao seu alcance e sobre a sua responsabilidade de coleta. O aplicativo processa os dados coletados e realiza a sincronização das informações com os participantes *Resident* da MCC. Assim, caso um *mobile* fique indisponível na MCC, as informações coletadas e processadas não são perdidas. Após o processo de coleta, processamento e sincronização, o *Header* atualiza o perfil de consumo no *WSInside* com as informações consolidadas. A consolidação das informações é baseada no somatório do consumo energético de cada categoria de *appliances*. A frequência de atualização é parametrizada para atender as demandas da concessionária de energia.

Os aplicativos participantes da MCC trocam mensagens IAA (*I Am Alive*) entre si para saber quem está ativo na MCC, e principalmente, para saber se o *Header* está ativo. Quando um aplicativo detectar que o *Header* está inativo, requisita ao *WSInside* que inicie uma eleição de *Header*. Adotamos eleição por escolha aleatória para evitar ataques, pois não é possível inferir quem será o *Header*

## 6.1 Teste de Substituição do *Header*

Neste teste é levado em consideração a mudança do *Header*, pois o residente pode se ausentar da casa e outro *mobile* deve assumir a função do *Header*.

A substuição do *Header* acontece de modo aleatório, pois considera-se um risco a segurança ter uma ordem específica na escolha, entende-se que se um atacante ter conhecimento de qual dispositivo será o próximo *Header*, ele pode-se realizar o ataque àquele *mobile* antes mesmo de se tornar *Header*.

Por questão de segurança apenas *Resident* poderá se tornar um *Header*, pois contêm todas as informações da casa, e seria inviável no ponto de vista de segurança da residência que alguém de fora, mesmo que confiável (*trust*), ter acesso a tais informações.

Desta forma foi considerado uma *smart house* com 05 moradores. Esses moradores foram adicionados no *WSInside* e receberam como credenciais do tipo *Resident,* podendo assim, tornar-se *Header*. Para a confirmação de presença os mobiles devem trocar mensagens

[15] http://developer.android.com/reference/android/telephony/package-summary.html
[16] https://www.sqlite.org

IAA entre sí e enviar também para o *WSInside*. O mecanismo de envio de IAA ao *WSInside* serve para o controle do *Header*, caso *WSInside* detecte sua ausência, dará início a eleição automática e aleatória do novo *Header*. Conforme observado na Figura 6.1, o *Header* é o número "+554399775658". A cada segundo é enviado entre os participantes da MCC e o *WSInside* a mensagem verificando se o *Header* ainda está ativo.

```
================================Ciclo de 5 segundos=========================================
Numero: +554399775658 | Presente: True | Header : True  | ResidentMember : True | Time life 2
Numero: +554384288682 | Presente: True | Header : False  | ResidentMember : True | Time life 2
Numero: +554384375968 | Presente: True | Header : False  | ResidentMember : True | Time life 2
Numero: +554399129155 | Presente: True | Header : False  | ResidentMember : True | Time life 2
Numero: +554384351239 | Presente: True | Header : False  | ResidentMember : True | Time life 2
Header : +554399775658
============================================================================================
```

Figura 6.1 – *Header* presente.

Após um ciclo de 5 segundos sem a presença do *Header* ativo, a eleição será realizada. A espera por 5 segundos se dá pelo motivo que o *mobile* do *Header* pode se afastar ou ficar fora do alcance de um roteador por alguns segundos dependendo da posição que ele estiver dentro da casa, e para não refazer o sistema de entrada na MCC qualquer mobile ativo pode se ausentar por este tempo determinado.

Por motivos de teste, desligou-se o *mobile* do *Header*, simulando sua saída casa. Conforme Figura 5.3, o *Resident* "+554399775658" não está mais presente na casa (ativo na MCC), então o novo *Header* foi eleito para assumir a função.

```
================================Ciclo de 5 segundos=========================================
Numero: +554399775658 | Presente: False | Header : False  | ResidentMember : True | Time life -2
Numero: +554384288682 | Presente: True | Header : False  | ResidentMember : True | Time life 3
Numero: +554384375968 | Presente: True | Header : False  | ResidentMember : True | Time life 3
Numero: +554399129155 | Presente: True | Header : True  | ResidentMember : True | Time life 3
Numero: +554384351239 | Presente: True | Header : False  | ResidentMember : True | Time life 3
Header : +554399129155
============================================================================================
```

Figura 6.2 - Troca de *Header*.

## 6.2 Avaliação

Para realizar a avaliação do protótipo, foi construído um cenário com 04 *mobiles*, suas configurações são mostradas na tabela 1.

**Tabela 6.1 - *Mobiles* utilizados na avaliação.**

| *Mobile* | Processor (GHz) | RAM (GB) | Android |
|---|---|---|---|
| Motorola G2 | Quad-core 1.2 | 1.0 | 5.02 |
| Samsung S3 mini | Dual-core 1.2 | 1.0 | 4.1 |
| Asus ZenFone 5 | Dual-core 1.2 | 2.0 | 4.3 |
| Nexus 7 | Quad-core 1.9 | 1.0 | 4.4.2 |

Fonte: Autoria própria.

O teste realizado tem como objetivo avaliar o comportamento da MCC com o aumento do número de *mobiles*. Neste teste foi desenvolvido um ambiente onde os *mobiles* coletam, processam, sincronizam e armazenam os dados de consumo dos *appliances*. Neste cenário definimos um *Header* (Motorola G2), com o intuito de não ter interferências nas medidas, devido a diferentes configurações de *hardware*. Definimos que na *smart house* há 12 *appliances* que fornecem 10 atributos de consumos por requisição. Então, fomos adicionando os demais *mobiles* para avaliar o impacto na performance da MCC (Figura 5). Como esperado, é possível observar que o aumento do número de *mobiles* na MCC melhora proporcionalmente o seu desempenho, devido a distribuição das tarefas de coleta, processamento e armazenamento.

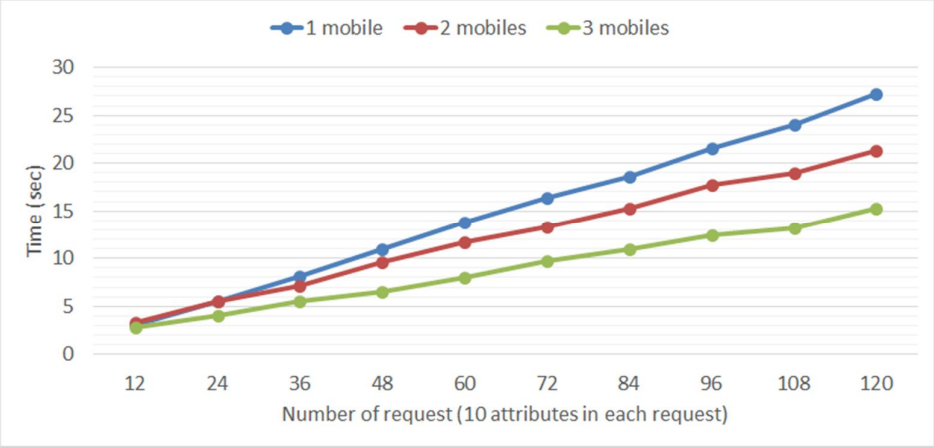


Figura 6.3 – Gráfico de avaliação do número de *mobiles* na MCC.

Os testes têm como objetivo avaliar o impacto do número de *mobiles* presentes na MCC e o tempo de sincronização entre eles. Para isso, foi variado o número de *mobiles* na MCC, número de requisições e o número de *appliances*.

O segundo teste foi realizado para verificar o *overhead* do número de participantes na MCC em relação a sincronização dos dados coletados. Deste modo, foi desconsiderado o tempo de coleta e processamento e apenas avaliado o tempo de sincronização das informações, tendo o *Header* como referência. Do mesmo modo que o teste anterior, o *Header* permaneceu o mesmo durante todo o teste. O tempo médio de sincronização por atributo, utilizando dois *mobiles* foi de 12.29 ms. Utilizando três *mobiles* na MCC, o tempo gasto foi 14.3 ms e para quatro *mobiles* 15.86 ms. Dessa forma, podemos observar que há um *overhead* na sincronização dos *mobiles* da MCC, de cerca de 10 a 15% para cada *mobile* adicionado.

# 7 Conclusão

Neste trabalho apresentamos um mecanismo de gestão do consumo energético de uma *smart house* com segurança e privacidade concebidas *by design*. Diferentemente da literatura, isto permite que a casa esteja acessível a partir da Internet, porém sem expor seus *appliances*. Além disto, utilizamos *ad hoc mobile cloud*, mecanismo descentralizado que aproveita o poder computacional dos *mobiles* de cada morador, sem exigir custos adicionais com compra de equipamentos específicos.

Nossa proposta tem uma arquitetura concebida para evitar os pontos únicos de falhas e comprometimento por ataques. Alterações nos perfis de consumo e cadastro de membros da MCC só são possíveis se o *mobile* estiver alcançável na LAN da *smart house*, isto evita ataques e violações de privacidade que venham da Internet. Este nível de segurança e privacidade é possível porque a MCC é formada *ad hoc* e o acesso direto aos *appliances* a partir da Internet não é possível. O esquema proposto isola a MCC da Internet, deixando-a acessível apenas dentro da *smart house*.

Para que a MCC tenha acesso a Internet, e.g. responder a solicitações da companhia de energia elétrica, foi criado um serviço que intermedia a comunicação. Assim, quando a companhia de energia elétrica necessita realizar a redução do consumo energético, por exemplo, para evitar um possível apagão, esta comunica-se com um serviço que serve exclusivamente para receber as solicitações externas. É necessário que outro serviço interno detecte as solicitações da companhia elétrica para que seja iniciado um processo de redução de consumo energético interno a *smart house*, que envolverá a leitura dos perfis de consumos e executado pela MCC.

Nosso protótipo mostra a viabilidade da nossa proposta, quando avaliamos o comportamento da MCC com o aumento do número de participantes. A análise mostra que aumentando o número de participantes da MCC, aumenta proporcionalmente a sua performance, apesar da sobrecarga na gestão do número de *mobiles*. Realizamos um teste para verificar a sobrecarga do número de participantes na MCC em relação a sincronização dos

dados coletados. Os resultados mostram que existe um *overhead* de aproximadamente 10 a 15% para cada *mobile* adicionado na MCC.

Como trabalho futuro, pretendemos utilizar *smart appliances* reais para realizar os testes, implementando o aplicativo na plataforma iOS e *Windows Phone*.

# Referências Bibliográficas

[1] A. R. Al-Ali e M. AL-Rousan, "Java-based home automation system", *IEEE Trans. Consum. Electron.*, vol. 50, nº 2, p. 498–504, maio 2004.

[2] A. Z. Alkar e U. Buhur, "An internet based wireless home automation system for multifunctional devices", *IEEE Trans. Consum. Electron.*, vol. 51, nº 4, p. 1169–1174, nov. 2005.

[3] M. Armbrust, A. Fox, A. Fox, R. Griffith, R. Griffith, A. Joseph, A. Joseph, RH, e RH, "Above the clouds: A Berkeley view of cloud computing", *Univ. California, Berkeley, Tech. Rep. UCB* , p. 07–013, 2009.

[4] L. Atzori, A. Iera, e G. Morabito, "The Internet of Things: A survey", *Comput. Networks*, vol. 54, nº 15, p. 2787–2805, 2010.

[5] R. Bhattacharjee, "An analysis of the cloud computing platform.", Massachusetts Institute of Technology, 2009.

[6] D. C. Brabham, "*Crowdsourcing"*, 2013.

[7] A. Chakravorty, T. Wlodarczyk, e Chunming Rong, "Privacy Preserving Data Analytics for Smart Homes", in *2013 IEEE Security and Privacy Workshops*, p. 23–27,2003.

[8] M. Chan, E. Campo, D. Estève, e J.-Y. Fourniols, "Smart homes — Current features and future perspectives", *Maturitas*, vol. 64, nº 2, p. 90–97, out. 2009.

[9] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, e D. Zeinalipour-Yazti, "Crowdsourcing with Smartphones", *IEEE Internet Comput.*, vol. 16, nº 5, p. 36–44, set. 2012.

[10] C.-C. Chung, C. Y. Huang, S.-C. Wang, e C.-M. Lin, "Bluetooth-Based Android Interactive Applications for Smart Living", in *2011 Second International Conference on Innovations in Bio-inspired Computing and Applications*, p. 309–312, 2011.

[11] D. J. Cook, "How Smart Is Your Home?", *Science (80)* vol. 335, nº 6076, p. 1579–1581, 2012.

[12] E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, e P. Bahl, "MAUI: Making Smartphones Last Longer with Code Offloa", in *Proceedings of the 8th*

*international conference on Mobile systems, applications, and services - MobiSys*, p. 49, 2010.

[13] M. Daro Kristensen, "Scavenger: Transparent development of efficient cyber foraging applications", in *2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, p. 217–226, 2010.

[14] Deepali Javale, Mohd. Mohsin, Shreerang Nandanwar, Mayur Shingate, "Home Automation and Security System Using Android ADK", *Int. J. Electron. Commun. Comput. Technol.*, p. 382–385, 2013.

[15] G. Demiris, B. K. Hensel, M. Skubic, e M. Rantz, "Senior residents' perceived need of and preferences for 'smart home' sensor technologies.", *Int. J. Technol. Assess. Health Care*, vol. 24, nº 1, p. 120–124, 2008.

[16] S. S. S. R. Depuru, L. Wang, e V. Devabhaktuni, "Smart meters for power grid: Challenges, issues, advantages and status", *Renew. Sustain. Energy Rev.*, vol. 15, nº 6, p. 2736–2742, 2011.

[17] S. Dihal, H. Bouwman, M. de Reuver, M. Warnier, e C. Carlsson, "Mobile cloud computing: state of the art and outlook", *info*, vol. 15, nº 1, p. 4–16, jan. 2013.

[18] A. Elshafee e K. A. Hamed, "Design and Implementation of a WiFi Based Home Automation System", *World Acad. Sci. Eng. Technol.* vol. 6, nº 8, p. 1856–1862, 2012.

[19] Z. Fan, P. Kulkarni, S. Gormus, C. Efthymiou, G. Kalogridis, M. Sooriyabandara, Z. Zhu, S. Lambotharan, e W. H. Chin, "Smart Grid Communications: Overview of Research Challenges, Solutions, and Standardization Activities", *IEEE Commun. Surv. Tutorials*, vol. 15, nº 1, p. 21–38, jan. 2013.

[20] N. Fernando, S. W. Loke, e W. Rahayu, "Mobile cloud computing: A survey", *Futur. Gener. Comput. Syst.*, vol. 29, nº 1, p. 84–106, 2013.

[21] V. J. Forte, "Smart Grid at National Grid", *Innov. Smart Grid Technol.* p. 1–4, 2010.

[22] S. D. Fugita, F. A. S. Borges, R. A. S. Fernandes, e I. N. Da Silva, "Methodology based on smart meters applied to the identification of residential loads", in *2012 4th Electronic System-Integration Technology Conference, ESTC*, 2012.

[23] H. Ghayvat, S. Mukhopadhyay, X. Gui, e N. Suryadevara, "WSN- and IOT-Based Smart Homes and Their Extension to Smart Buildings", *Sensors*, vol. 15, nº 5, p. 10350–10379, 2015.

[24] M. G. Golzar e H. Tajozzakerin, “A New Intelligent Remote Control System for Home Automation and Reduce Energy Consumption”, in *2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, p. 174–180, 2010.

[25] D. Guinard e V. Trifa, “Towards the Web of Things: Web Mashups for Embedded Devices”, *Work. Mashups, Enterp. Mashups Light. Compos. Web (MEM 2009), Proc. WWW (International World Wide Web Comf)*, p. 1–8, 2009.

[26] J. Han, H. Lee, e K.-R. Park, “Remote-controllable and energy-saving room architecture based on ZigBee communication”, *IEEE Trans. Consum. Electron.* vol. 55, nº 1, p. 264–268, fev. 2009.

[27] Haowen Chan e A. Perrig, “Security and privacy in sensor networks”, *Computer (Long. Beach. Calif)*, vol. 36, nº 10, p. 103–105, out. 2003.

[28] S. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, e E. Jansen, “The Gator Tech Smart House: a programmable pervasive space”, *Computer (Long. Beach. Calif)*, vol. 38, nº 3, p. 50–60, mar. 2005.

[29] A. U. R. Khan, M. Othman, S. A. Madani, e S. U. Khan, “A Survey of Mobile Cloud Computing Application Models”, *IEEE Commun. Surv. Tutorials*, vol. 16, nº 1, p. 393–413, jan. 2014.

[30] H. Khurana, M. Hadley, Ning Lu, e D. A. Frincke, “Smart-grid security issues”, *IEEE Secur. Priv. Mag.*, vol. 8, nº 1, p. 81–85, jan. 2010.

[31] I. Korkmaz, S. K. Metin, A. Gurek, C. Gur, C. Gurakin, e M. Akdeniz, “A cloud based and Android supported scalable home automation system”, *Comput. Electr. Eng.*, vol. 43, p. 112–128, 2015.

[32] K. Kumar, J. Liu, Y. H. Lu, e B. Bhargava, “A survey of computation offloading for mobile systems”, *Mob. Networks Appl.*, vol. 18, nº 1, p. 129–140, 2013.

[33] M. Kuzlu, M. Pipattanasomporn, e S. Rahman, “Communication network requirements for major smart grid applications in HAN, NAN and WAN”, *Comput. Networks*, vol. 67, p. 74–88, jul. 2014.

[34] H. Lamin, “Análise de impacto regulatório da implantação de redes inteligentes no Brasil”, 2013.

[35] Q. Li e M. Zhou, “The Future-Oriented Grid-Smart Grid”, *J. Comput.*, vol. 6, nº 1, p. 98–105, jan. 2011.

[36] K. Y. Lian, S. J. Hsiao, e W. T. Sung, “Intelligent multi-sensor control system based on innovative technology integration via ZigBee and Wi-Fi networks”, *J. Netw. Comput. Appl.*, vol. 36, nº 2, p. 756–767, 2013.

[37] N.-S. Liang, L.-C. Fu, e C.-L. Wu, “An integrated, flexible, and Internet-based control architecture for home automation system in the Internet era”, in *Proc. of the International Conference on Robotics and Automation*, vol. 2, p. 1101–1106, 2002.

[38] S.-W. Luan, J.-H. Teng, S.-Y. Chan, e L.-C. Hwang, “Development of a smart power meter for AMI based on ZigBee communication”, in *2009 International Conference on Power Electronics and Drive Systems (PEDS)*, p. 661–665, 2009.

[39] E. Marinelli, “Hyrax: Cloud Computing on Mobile Devices using MapReduce”, Carnegie Mellon University, 2009.

[40] P. McDaniel e S. McLaughlin, “Security and privacy challenges in the smart grid”, *IEEE Secur. Priv.*, vol. 7, nº 3, p. 75–77, 2009.

[41] V. Della Mea, E. Maddalena, e S. Mizzaro, “Crowdsourcing to Mobile Users: A Study of the Role of Platforms and Tasks”, in *Proc. of DBCrowd - VLDB Workshop on Databases and Crowdsourcing*, p. 14–19, 2013.

[42] T. Mell e P. Grance, *“Working Definition of Cloud Computing”*, *Draft NIST*, vol. 53, 2009.

[43] F. Moraes, A. Amory, N. Calazans, E. Bezerra, e J. Petrini, “Using the CAN protocol and reconfigurable computing technology for Web-based smart house automation”, *Symp. Integr. Circuits Syst. Des.*, 2001.

[44] B. Mrazovac, M. Z. Bjelica, N. Teslic, e I. Papp, “Towards ubiquitous smart outlets for safety and energetic efficiency of home electric appliances”, in *2011 IEEE International Conference on Consumer Electronics -Berlin (ICCE-Berlin)*, p. 322–326, 2011.

[45] S. C. Patel e P. Sanyal, “Securing SCADA systems”, *Inf. Manag. Comput. Secur*, vol. 16, nº 4, p. 398–414, out. 2008.

[46] G. Patrício e L. Gomes, “Smart house monitoring and actuating system development using automatic code generation”, in *IEEE International Conference on Industrial Informatics (INDIN)*, 2009, p. 256–261.

[47] C. Perera, P. P. Jayaraman, A. Zaslavsky, D. Georgakopoulos, e P. Christen, “MOSDEN: An Internet of Things Middleware for Resource Constrained Mobile Devices”, in *2014 47th Hawaii International Conference on System Sciences*, p. 1053–1062, 2014.

[48] C. Perera, P. P. Jayaraman, A. Zaslavsky, D. Georgakopoulos, e P. Christen, “Sensor discovery and configuration framework for the Internet of Things paradigm”, *2014 IEEE World Forum Internet Things*, p. 94–99, 2014.

[49] C. W. Potter, A. Archambault, e K. Westrick, “Building a smarter smart grid through better renewable energy information”, in *2009 IEEE/PES Power Systems Conference and Exposition*, p. 1–5, 2009.

[50] J. Potts e S. Sukittanon, “Exploiting Bluetooth on Android mobile devices for home security application”, in *2012 Proceedings of IEEE Southeastcon*, p. 1–4, 2012.

[51] A. Rajabzadeh, A. Manashty, e Z. Jahromi, “A Mobile Application for Smart House Remote Control System”, *Proc. ICWCMC - Int. Conf. Wirel. Commun. Mob. Comput.*, p. 80–86, 2010.

[52] N. M. Rao, “Cloud Computing Through Mobile-Learning”, *Int. J.*, vol. 1, nº 6, p. 42–47, 2010.

[53] M. Satyanarayanan, P. Bahl, R. Caceres, e N. Davies, “The Case for VM-Based Cloudlets in Mobile Computing”, *IEEE Pervasive Comput.*, vol. 8, nº 4, p. 14–23, out. 2009.

[54] U. Sharma e S. R. N. Reddy, “Design of Home/Office Automation using Wireless Sensor Network”, *Int. J. Comput. Appl.*, vol. 43, nº 22, p. 46–52, 2012.

[55] M. H. Shwehdi e A. Z. Khan, “A power line data communication interface using spread spectrum technology in home automation”, *Power Deliv. IEEE Trans.*, vol. 11, nº 3, p. 1232–1237, 1996.

[56] U.S. Department of Energy, “the SMART GRID”, 2010.

[57] L. M. Vaquero, L. Rodero-Merino, J. Caceres, e M. Lindner, “A break in the clouds”, *ACM SIGCOMM Computer Communication Review*, vol. 39, nº 1. p. 50, 2008.

[58] K. I. K. Wang, W. H. Abdulla, e Z. Salcic, “Ambient intelligence platform using multi-agent system and mobile ubiquitous hardware”, *Pervasive Mob. Comput.*, vol. 5, nº 5, p. 558–573, 2009.

[59] S. D. Warren e L. D. Brandeis, “The Right to Privacy”, *Harv. Law Rev.*, vol. 4, nº 5, p. 193, dez. 1890.

[60] M. Weiss e D. Guinard, “Increasing energy awareness through web-enabled power outlets”, in *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia - MUM ’10*, p. 1–10, 2010.

[61] A. Witkovski, A. Santin, V. Abreu, e J. Marynowski, “An IdM and Key-based Authentication Method for providing Single Sign-On in IoT.”, *IEEE* GLOBECOM, 2015, San Diego, USA, 2015.

[62] M. Yan e H. Shi, “Smart Living Using Bluetooth-Based Android Smartphone”, *Int. J. Wirel. Mob. Networks*, vol. 5, nº 1, p. 65–72, fev. 2013.

[63] Y. Yan, Y. Qian, H. Sharif, e D. Tipper, “A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges”, *IEEE Commun. Surv. Tutorials*, vol. 15, nº 1, p. 5–20, jan. 2013.

[64] S. Zeadally, A.-S. K. Pathan, C. Alcaraz, e M. Badra, “Towards Privacy Protection in Smart Grid”, *Wirel. Pers. Commun.*, vol. 73, nº 1, p. 23–50, nov. 2013.

[65] Q. Zhang, L. Cheng, e R. Boutaba, “Cloud computing: State-of-the-art and research challenges”, *J. Internet Serv. Appl.*, vol. 1, nº 1, p. 7–18, 2010.