ANÁLISE DE VULNERABILIDADES EM IOT
Um estudo de caso comparando Blynk e Cayenne

Denis Pohlmann Gonçalves[1], **Anderson Monteiro da Rocha**[1], **Gleizer Bierhalz Voss**[1], **Henrique Tamiosso Machado**[1]

[1]Instituto Federal Farroupilha - Campus São Vicente do Sul (IFFar-SVS)
Rua 20 de setembro, 2026 - São Vicente do Sul/RS - Brasil

{denis.goncalves,anderson.monteiro}@iffarroupilha.edu.br

{gleizer.voss, henrique.tamiosso}@iffarroupilha.edu.br

Abstract. *With the increasing number of smart devices on the Internet, the technological paradigm of the Internet of Things brings numerous challenges involving various issues, with safety as a major concern. Numerous threats are encountered daily by network administrators and ethical hackers. In this way, through a case study, this research aims to analyze the vulnerabilities of some free IoT applications for smartphones, the IoT device and the communication between them, through some specific tools. With the tests, a critical vulnerability has been identified in the communication between a mobile application and its IoT device, posing serious security risks for the user using such a solution.*

Resumo. *Com o crescente aumento de dispositivos inteligentes na Internet, o paradigma tecnológico da Internet das Coisas traz consigo inúmeros desafios envolvendo variadas questões, tendo a segurança como grande preocupação. Inúmeras ameaças são encontradas diariamente por administradores de rede e hackers éticos. Dessa maneira, através de estudo de caso, esta pesquisa tem como objetivo analisar as vulnerabilidades de alguns aplicativos IoT gratuitos para smartphones, o dispositivo IoT e a comunicação entre eles, por meio de algumas ferramentas específicas. Com os testes, identificou-se uma vulnerabilidade crítica na comunicação entre um aplicativo mobile e seu dispositivo IoT, trazendo sérios riscos de segurança para o usuário que utiliza tal solução.*

1. Introdução

A Internet das Coisas (IoT, do inglês *Internet of Things*), está mudando a forma de como interagimos com o mundo ao nosso redor. Inúmeros dispositivos que no passado realizavam somente suas funções básicas, agora estão recebendo funcionalidades que possibilitam a comunicação em rede com outros equipamentos, resultando em dispositivos inteligentes, como eletrodomésticos, veículos, prédios, roupas, dentre outros. Apesar da IoT estar em alta no momento, equipamentos inteligentes já fazem parte do nosso cotidiano desde 2010, como *smartvs* e *smartphones*, tendo em seu mercado global um crescimento previsto para US$ 12,8 bilhões entre 2014 a 2020 [Lund et al. 2014].

Com a crescente demanda de dados trafegando pela rede, é imprescindível a segurança e privacidade na comunicação entre os dispositivos IoT, de forma a garantir os princípios da segurança da informação. Toda a informação tem o seu ciclo de vida, que

é compreendido desde sua criação até o descarte, apresentando vários momentos que a colocam em risco [Sêmola 2013]. Sendo assim, o problema que se apresenta é: quais são as possíveis vulnerabilidades que poderiam ser encontradas nas aplicações *mobile* gratuitas e sua comunicação com dispositivos IoT? Como hipótese, pode ser considerada a possibilidade de haver falhas de segurança, como *backdoors* e dados não criptografados.

O interesse por esse assunto surgiu com o aumento significativo de dispositivos IoT personalizáveis, construídos com *hardware* livre de baixo custo aliados a aplicações de uso gratuito, usados principalmente para automação residencial e afins. Esta pesquisa pode ser caracterizada quanto a sua natureza como qualitativa e quanto a sua finalidade como exploratória. A coleta de dados foi realizada por meio de estudo de caso, tendo como foco a comparação entre as aplicações *mobile* de IoT: Blynk[1] e Cayenne[2].

2. Arquitetura IoT

A IoT pode ser caracterizada como uma rede de objetos, veículos, prédios e outros que possuam tecnologia embarcada, equipados com sensores e conexão, capazes de coletar e transmitir dados [Xia et al. 2012]. Esta tecnologia vem sendo popularizada mundialmente permitindo a comunicação em rede de equipamentos dos mais variados tipos.

Para que um "objeto inteligente" seja conectado na Internet, deve-se ter uma arquitetura escalável e interoperável, permitindo trabalho com diversos sensores e operação em variadas tecnologias de comunicação, respectivamente. A literatura apresenta variadas pesquisas e propostas de arquiteturas, desde as básicas até as sofisticadas, que baseiam-se nas necessidades acadêmicas e industriais [Al-Fuqaha et al. 2015].

Em um modelo básico de arquitetura em três camadas, a primeira camada, denominada "Camada de Percepção", representa os objetos inteligentes. Esta camada concentra os objetos físicos com o objetivo de interagir com o mundo real, compostos de microcontroladores, sensores, atuadores, dentre outros, conforme a função desejada. A "Camada de Rede", representa a abstração das tecnologias de rede utilizadas para efetivar a comunicação, sendo, alguns serviços de gerenciamento, *gateways* e processos de identificação. A terceira e última camada, denominada "Camada de Aplicação", representa todos os serviços providos ao usuário ou cliente. Nesta camada estão os *softwares* e dispositivos capazes de enviar e receber informações com os dispositivos da primeira camada. Todos os modelos, mesmo os mais sofisticados, possuem as camadas do modelo básico visto que este contém os componentes mínimos necessários de funcionamento.

2.1. Segurança

A Segurança da informação é extremamente importante e essencial em ambientes computacionais tornando sua utilização um elemento crítico na atualidade. Apesar da simplicidade e funcionalidade que a tecnologia de IoT pode trazer, ela também carrega consigo ameaças à segurança dos usuários. Ao utilizar um objeto inteligente, o mesmo encontra-se suscetível a certos riscos e ameaças, visto que encontra-se conectado à Internet.

A segurança em dispositivos IoT pode ser implementada através da criptografia em diferentes camadas da pilha de protocolos TCP/IP. Por exemplo, na camada de enlace,

[1]Disponível em: https://www.blynk.cc/

[2]Disponível em: https://mydevices.com/cayenne/landing/arduino/

poderia ser implementado o *Advanced Encryption Standard* (AES), contudo não haveria integridade nos dados. Na camada de rede, poderia ser utilizado o mecanismo IPSec que tem como recurso principal encriptar e/ou autenticar todo o tráfego a nível IP, protegendo todas as informações [Stallings 2015]. Entretanto, utilizar esse mecanismo consumiria grande parte dos recursos de *hardware* em um dispositivo IoT.

2.2. Vulnerabilidade

As vulnerabilidades em tecnologia da informação podem ser oriundas de erros no projeto ou no seu desenvolvimento e falhas nos equipamentos tanto em *hardware* como *software*. Essas vulnerabilidades podem ser exploradas por atacantes através de práticas maliciosas como invasão, para acessar informações confidenciais ou ataques de negação de serviço tornando serviços inacessíveis [Nakamura and de Geus 2007].

A análise de vulnerabilidades possibilita aos analistas, administradores de rede ou mesmo usuários com conhecimentos aprofundados (*pentesters*), identificar as possíveis vulnerabilidades e tratá-las antes que tornem-se um problema. Esta ação reativa praticada por um *pentester*, pode eliminar ou ao menos mitigar a janela de exposição a uma possível ameaça, evitando a perda de informações importantes [Melo 2017].

2.3. Ataque de Rede

Muitos riscos estão envolvidos na utilização da IOT, caso não sejam aplicadas medidas de segurança. Os ataques aumentam na mesma proporção em que a tecnologia se expande. Segundo a Abranet [ABRANET 2012], no ano de 2016 os ataques de negação de serviço (DDoS) aumentaram cerca de 138% em relação ao ano anterior, tendo como origem dos ataques, dispositivos IoT infectados, fazendo parte de *botnets*. Como exemplo, pode ser citado a falha de segurança encontrada em um Jeep Cherokee no ano de 2015, onde dois pesquisadores exploraram vulnerabilidades no veículo, obtendo o controle remoto de muitas de suas funções, como desligar o motor, controlar a aceleração e frenagem [Greenberg 2015]. Esta falha apresenta sérios riscos à vida dos condutores e passageiros e, se explorada por cibercriminosos, poderia acarretar em consequências fatais.

3. Aplicativos e dispositivo IoT

Apesar de existirem vários aplicativos capazes de controlar dispositivos IoT genéricos, para este trabalho foram selecionados o Blynk e o Cayenne. A escolha desses *apps* se deu pelo fato de serem gratuitos e com as melhores avaliações do público tanto na *Play Store* quanto na *App Store*.

Com relação ao Blynk, pode ser caracterizado como uma plataforma com aplicativos para Android e IOS capaz de controlar *hardwares* livres como Arduino, Raspberry Pi, dentre outros. Com o Blynk é possível criar um projeto de IoT de maneira muito simples, apenas arrastando objetos, sem precisar programar uma linha sequer de código, proporcionando uma forma muito fácil de criar as suas redes, mesmo para aqueles usuários sem conhecimentos em programação.

Já o Cayenne, é um criador de projetos IoT considerado o primeiro no mundo com as funções de arrastar e soltar os seus componentes, não necessitando de codificação em projetos simples. Também é compatível com diversos dispositivos e opções de conectividade, contando com sua própria API MQTT, e variados componentes que permitem o monitoramento e controle remoto de diversos dispositivos.

Quanto ao dispositivo, foi utilizado o microcontrolador ESP8266. Ele é composto de um conjunto completo de processamento, pinos de entrada e saída e suporte a vários protocolos de comunicação, com destaque a WiFi. Por unificar todas essas funcionalidades em um único dispositivo, justifica sua utilização em projetos para os mais variados fins, desde automação de dispositivos como eletrodomésticos e veículos, até projetos educacionais em ambientes escolares e acadêmicos, como em robótica e eletrônica.

4. Implementação e Testes

A partir de um estudo preliminar através da análise do cenário de testes, identificaram-se as áreas de maior vulnerabilidade. Assim, foram definidos como pontos de análise, aqueles locais identificados como críticos, isto é, aqueles pontos onde havia a possibilidade da informação ser acessada/interceptada. Para isso, foram consideradas situações como: acesso indevido ao dispositivo IoT; interceptação das informações na comunicação entre o dispositivo e o aplicativo e; a intrusão no próprio aplicativo. A realização dos testes se deu em três etapas distintas.

Para que o dispositivo IoT utilizado nesta pesquisa pudesse ser analisado, primeiro realizou-se o *upload* do código necessário a fim de que o dispositivo pudesse estabelecer conexão com a sua aplicação. Para isso, utilizou-se um ambiente de desenvolvimento integrado, do inglês *Integrated Development Environment* (IDE), agilizando o processo do desenvolvimento do código. Para cada aplicação, Blynk e Cayenne, utilizou-se como base o código de exemplo disponível em suas bibliotecas versões 0.4.10 e 1.0.1 respectivamente, adaptando-o para conexão na rede local, acionamento do *led* embutido na placa do microcontrolador e permissão da leitura de uma de suas portas.

Em cada dispositivo IoT, com os códigos já carregados, efetuou-se os testes com a utilização das ferramentas Sparta [3], OpenVas [4] e Nessus [5], tendo suas instalações em um sistema operacional Kali Linux 2018.1-amd64. A Figura 1 ilustra a topologia utilizada nos testes.

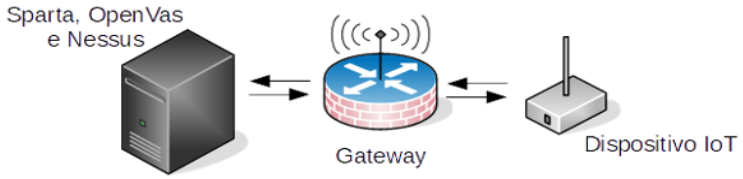

Figura 1. Topologia utilizada nos testes com o dispositivo IoT
Fonte: Dos autores

Por ser mais simples e consequentemente realizar testes mais rápidos, primeiramente utilizou-se a ferramenta Sparta. Nesta ferramenta, configurou-se os endereços IPv4 de ambos os dispositivos IoT como *hosts* alvos, optando-se pela versão de escaneamento completo e iniciando-se os testes na sequência. O objetivo dessa varredura é buscar portas abertas TCP ou UDP de serviços conhecidos, *backdoors* e coletar informações sobre o *hardware* e *software* que venham a contribuir para o objetivo do escaneamento.

[3]Disponível: https://sparta.secforce.com/
[4]Disponível: http://www.openvas.org/
[5]Disponível: http://www.software.com.br/c/fabricantes/tenable-network-security

Após as operações com o Sparta, iniciou-se os testes com a ferramenta OpenVas. Para isso, foi necessário criar uma tarefa de varredura para cada dispositivo IoT, considerando as configurações padrão da ferramenta. Na sequência, teve-se início as varreduras.

Com a finalização dos testes realizados no OpenVas, deu-se início aos procedimentos com a ferramenta Nessus. Após inicializar a ferramenta, definiu-se a política de varredura de *host*, sendo a mais adequada para o fim desta pesquisa. Na sequência, foram criadas e executadas as varreduras para os dois dispositivos IoT, identificados através de seus endereços IPv4.

As ferramentas OpenVas e Nessus, possuem o mesmo objetivo com as suas varreduras: buscar vulnerabilidades. Entretanto, efetuar os testes com as duas ferramentas mostra-se mais eficiente, uma vez que as varreduras são realizadas contando com a tecnologia e base de dados de cada ferramenta, oportunizando possíveis diferentes resultados.

Além disso, com o objetivo de fazer a inspeção nos pacotes de rede entre a aplicação e o dispositivo IoT em busca de possíveis vulnerabilidades, utilizou-se a ferramenta tcpdump versão 4.5.1, instalada no *gateway* da rede local com a Internet. A utilização da ferramenta nesse ponto da rede permite a coleta total de todos os pacotes IPv4 que trafegam na comunicação.

A captura dos pacotes foi definida em dois momentos, conforme a rota do tráfego gerado, primeiro: informações originadas nos aplicativos Blynk e Cayenne destinadas aos dispositivos IoT com seus respectivos códigos; segundo: informações originadas nos dispositivos IoT destinadas aos aplicativos. Em ambas as rotas o tráfego sempre é enviado ao *broker*, para que ele as encaminhe ao destino. A Figura 2 ilustra as rotas em que foram coletados os pacotes IPv4 e o ponto de coleta através do *sniffer* tcpdump.

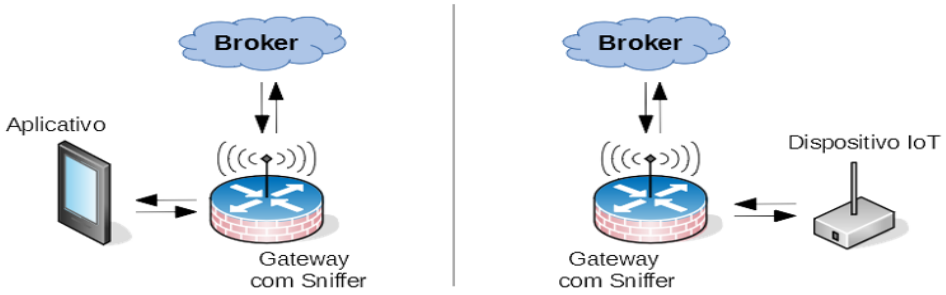

Figura 2. Rotas em que foram coletados os pacotes IPv4
Fonte: Dos autores

Após a coleta dos pacotes, com a ajuda da ferramenta Wireshark somente para visualização, realizou-se a inspeção detalhada de todos os pacotes nas duas rotas. Em cada pacote, examinou-se as camadas de rede buscando falhas de segurança, como a utilização de protocolos vulneráveis ou informações não criptografadas.

5. Resultados e Discussões

Nesta seção são apresentados os resultados de todos os testes efetuados em busca de características que revelem as possíveis vulnerabilidades encontradas. Para cada teste, foram executados três vezes, apresentando sempre o mesmo resultado. Algumas informações são mostradas através de análise comparativa das soluções estudadas na seção 3, na

forma de tabelas. A partir da análise realizada nos dispositivos IoT com as ferramentas Sparta, OpenVas e Nessus pode-se identificar as características a seguir.

No Sparta, os resultados para ambos os dispositivos IoT foram semelhantes, tendo como mudança, já esperada, apenas o endereço físico do dispositivo. Após a varredura de portas disparada, não foram encontradas portas disponíveis para conexão, bem como portas de serviços e *backdoors*. Algumas outras informações foram apresentadas pela ferramenta como, estado do dispositivo, mac *address*, sistema operacional e precisão do teste. Na tabela 1, são mostradas as informações resultantes dos testes efetuados com ambos dispositivos pela ferramenta Sparta. Os endereços físicos e de rede foram suprimidos devido a não relevância.

Dispositivo IoT	Protocolo	Portas			Sistema operacional	Precisão	Tempo
		abertas	fechadas	filtradas			
com código Blynk	TCP e UDP	0	65535	0	NodeMCU firmware (lwIP stack)	95	93s
com código Cayenne	TCP e UDP	0	65535	0	NodeMCU firmware (lwIP stack)	95	88s

Tabela 1. Resultado dos testes realizados com o Sparta

Conforme resultados coletados com a ferramenta Sparta, é possível constatar que não há portas abertas nos protocolos TCP e UDP, confirmando não haver vulnerabilidades nesse aspecto.

Os resultados obtidos dos testes realizados com o OpenVas e Nessus, não identificaram nenhuma vulnerabilidade nos dispositivos IoT avaliados, sendo gerados apenas registros das informações referentes ao teste. As duas ferramentas realizaram traçado de rota para o dispositivo, no qual obtiveram somente um pulo até o *host*, e identificaram o *fully qualifield domain name* (FQDN) do dispositivo, obtido através de consulta no servidor DNS local.

Com base nos resultados alcançados com as ferramentas Sparta, OpenVas e Nessus, pode-se garantir a segurança na utilização dos dispositivos IoT com códigos Blynk e Cayenne, bibliotecas versões 0.4.10 e 1.0.1 respectivamente, não havendo vulnerabilidades que possam causar danos ao usuário.

Na questão da comunicação, após a inspeção detalhada dos pacotes capturados, observou-se que as aplicações na sua versão nativa não utilizam criptografia, sendo que os dados trafegam em texto puro na comunicação entre dispositivo-broker-aplicativo. Entretanto, a solução Blynk utiliza protocolo próprio de comunicação, não sendo possível ler o conteúdo da camada de aplicação dos pacotes devido a ferramenta Wireshark não conter o protocolo em questão.

A Figura 3 mostra as informações visíveis em texto puro da solução Cayenne, como "*Client ID*", "*User Name*" e "*Password*" (identificação do cliente, nome de usuário e senha), referentes ao pacote em que o dispositivo IoT faz sua autenticação no *broker*.

Apesar dos pacotes serem capturados em rotas diferentes, as mesmas caracterís-

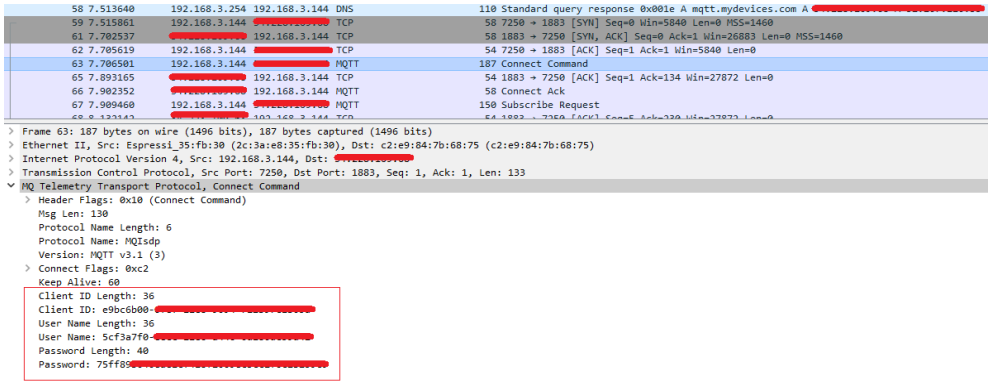

Figura 3. Informações visíveis em texto puro
Fonte: Dos autores

ticas foram identificadas nas soluções Blynk e Cayenne, antes e depois do *broker*, não havendo troca de protocolos ou adição de tecnologias. As características referentes a comunicação e segurança das informações identificadas na inspeção dos pacotes são exibidas na tabela 2.

Aplicação mobile – broker em nuvem	Protocolo utilizado na comunicação	Criptografia em instalação nativa	Leitura de informações
Blynk v. 2.18.1	Próprio	Não	Não foi possível ler
Cayenne v. 1.3.3-89	MQTT v. 3.1	Não	Texto puro

Tabela 2. Características identificadas com a inspeção de pacotes

Conforme verificado na inspeção dos pacotes, as soluções analisadas apresentam grande problema de segurança dos dados quando trafegados pela Internet, oferecendo riscos aos usuários, como roubo de informações, clonagem de conta, sabotagem nos dados e controle dos dispositivos IoT por algum intruso. Comparando as questões de segurança em relação as informações trafegarem em texto puro, percebe-se que a aplicação Cayenne aponta maior vulnerabilidade em relação a Blynk.

Contudo, para aplicações Blynk, é possível criar o seu próprio *broker*, nesse caso um Blynk server2. Com ele é possível estabelecer conexões criptografadas entre o aplicativo e o dispositivo IoT através de *Secure Sockets Layer* (SSL), resolvendo assim o problema da segurança.

Na questão dos aplicativos IoT estudados, Blynk e Cayenne, com base nos resultados das ferramentas SRT AppScanner e Malwarebytes, não foram encontradas vulnerabilidades. Essa situação demonstra segurança satisfatória, entretanto, os aplicativos ainda podem estar vulneráveis a alguma ameaça não contemplada pelas ferramentas de verificação utilizadas nos testes.

6. Considerações Finais

Esta pesquisa teve por finalidade apresentar, testar e comparar, nas questões de segurança e comunicação, dois aplicativos IoT para dispositivos móveis, bem como o *firmware* car-

regado no *hardware* específico utilizado para a comunicação com o *app mobile*. Os testes foram executados separadamente em três etapas, definidas com base em pontos onde a informação poderia ser acessada por alguém, podendo ser no próprio dispositivo IoT, no aplicativo IoT ou ainda na comunicação de rede entre ambos. Nos testes com os dispositivos e aplicações não foram encontradas vulnerabilidades, obtendo resultados muito semelhantes, divergindo apenas no tempo total de execução.

Já nos testes com a comunicação, foi encontrado uma "vulnerabilidade crítica", ou seja, que trás sérios riscos ao usuário que utiliza a aplicação Cayenne, confirmando a hipótese problema desta pesquisa, tendo suas informações trafegando em texto puro na Internet, permitindo ser facilmente explorada por um atacante mal intencionado. A aplicação Blynk também não utiliza criptografia, porém, emprega um protocolo próprio na comunicação inviabilizando a leitura das informações, mostrando maior segurança em comparação a aplicação Cayenne.

Como trabalhos futuros, seguindo a linha de IoT, seria relevante realizar dois estudos. Primeiro: pesquisar detalhadamente o protocolo MQTT, considerando seu funcionamento, características e versões, tendo foco a criptografia das informações. Segundo: pesquisar, implementar e avaliar as variadas arquiteturas para dispositivos IoT propostas pela comunidade acadêmica e empresas, realizando estudo comparativo de segurança e desempenho entre elas.

Referências

ABRANET (2012). Internet das Coisas faz ataques DDoS crescem 138% no Brasil. Disponível em: <https://www.ppgia.pucpr.br/ jamhour/Pessoal/Mestrado/TARC/QoSIP.pdf>. Acesso: agosto de 2016.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., and Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4):2347–2376.

Greenberg, A. (2015). Hackers Remotely Kill a Jeep on the Highway: With Me in It. Disponível em: <https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/>. Acesso: fevereiro de 2018.

Lund, D., MacGillivray, C., Turner, V., and Morales, M. (2014). Worldwide and regional internet of things (iot) 2014–2020 forecast: A virtuous circle of proven value and demand. *International Data Corporation (IDC), Tech. Rep*, 1.

Melo, S. (2017). *Exploração de Vulnerabilidades em Redes TCP/IP*. 3. Ed Rio de Janeiro: Alta Books.

Nakamura, E. T. and de Geus, P. L. (2007). *Segurança de redes em ambientes cooperativos*. Novatec Editora.

Sêmola, M. (2013). Gestão da segurança da informação: uma visão executiva. *Rio de Janeiro: Editora Elsevier 2013*.

Stallings, W. (2015). Criptografia e segurança de redes: principios e praticas. *6. ed. São Paulo: Pearson Education do Brasil*.

Xia, F., Yang, L. T., Wang, L., and Vinel, A. (2012). Internet of things. *International Journal of Communication Systems*, 25(9):1101–1102.