IFPI - FMN - UESPI

ENUCOMP

ENCONTRO UNIFICADO DE COMPUTAÇÃO EM PARNAÍBA

FACVLDADE MAURÍCIO DE NASSAU

FAZENDO PARTE DA SUA HISTÓRIA

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PIAUÍ

codiworks.com.br

VII Encontro Unificado de Computação

ENUCOMP 2014

12, 13 e 14 de Novembro

Anais Eletrônicos Enucomp 2014

Organização:

Thiago Carvalho de Sousa

Rodrigo Augusto Rocha Souza Baluz

1ª Edição

Editora: FUESPI

Informações:
www.**enucomp**.com.br

Patrocinadores :

Codiworks

TDA INFORMÁTICA

ingeti

FAPEPI

foxinline

deltanetwork

Apoio:

NUPEC DELTA

Parnaíba - Piauí

2014

www.enucomp.com.br/2014

Organização:
Thiago C. de Sousa
Rodrigo Augusto R. S. Baluz

Edição: 1ª
Editora: FUESPI

REALIZAÇÃO:

Parnaíba – Piauí
2014

Ficha Catalográfica elaborada pela Bibliotecária
Christiane Maria Montenegro Sá Lins CRB/3 - 952

E56a

ENCONTRO UNIFICADO DE COMPUTAÇÃO EM PARNAÍBA. (7: 2014: Parnaíba, PI).

Anais do VII ENUCOMP 2014, Parnaíba, PI, 12 a 14 de novembro de 2014: [recurso eletrônico]/ Organização [de] Thiago C. de Sousa e Rodrigo Augusto R. S. Baluz. - Parnaíba: FUESPI, 2014.

142 p.: Il.

ISBN: 978-85-8320-073-4

1. Ciência da Computação. 2. Congressos. I. Sousa, Thiago C. de (org.) II. Baluz, Rodrigo Augusto R. S. (org.) III. Título. CDD 001.642

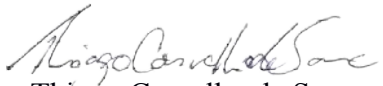
PREFÁCIO

Em 2008, as instituições parnaibanas que atuam no ensino superior, técnico e profissional de informática sentiram a necessidade de criar um evento de computação maior e mais completo para a cidade. Assim, surgiu o projeto ENUCOMP (Encontro Unificado de Computação em Parnaíba), criado numa parceria do CEEP, FAP, IFPI e UESPI, cujas propostas foram pautadas na contribuição para a troca de experiências, buscando a união dos acadêmicos; no fortalecimento da parceria no desenvolvimento da educação e da informática; e no incentivo à produção de trabalhos científicos ligados à área de tecnologia. Em sua sétima edição, apesar da ausência do CEEP, que desistiu da parceria por falta de incentivo de sua administração, o evento vem mantendo estes mesmos ideais, alcançando um crescimento consistente ano após ano e ganhando envergadura. Cerca de 300 participantes vindos de diversos Estados brasileiros devem comparecer ao ENUCOMP 2014, o que indica que já somos uma referência regional no Norte-Nordeste do Brasil na área de computação.

A edição deste ano, que ocorre nos dias 12, 13 e 14 de novembro, inclui cinco palestras, com temas sobre redes cognitivas, controle de veículos aéreos, big data, nanotecnologia e computação bioinspirada. Além disso, o evento oferece seis mini-cursos com assuntos relativos à modelos de negócios, desenvolvimento para Android, testes de invasão, usabilidade, automação com Arduíno e mineração de opiniões. A programação do ENUCOMP 2014 inovou mais uma vez em sua vertente científica ao co-alocar o I Workshop Piauiense de Pós-graduação em Ciências Computacionais, bem como montar um excelente Comitê de Programa, composto por quase cinquenta pesquisadores e professores de renomadas universidades e empresas de todas as cinco regiões brasileiras, para realizar a arbitragem de artigos por pares. Como consequência, foram recebidos 19 trabalhos, sendo selecionados apenas 8 para apresentação no evento, o que significou um taxa de aceitação de 42%, nível dos melhores congressos nacionais da área. Os artigos versam sobre três grandes áreas da computação: Inteligência Artificial, com trabalhos sobre testes de agentes, algoritmos genéticos para estimação de sistemas de potência, mineração de dados; Redes de Computadores, com temas relacionados à sensores em redes sem fio, segurança de sistemas móveis, aplicações NFC, balanceamento de carga; e Informática na Educação, com um assunto referente à produção de jogos educacionais. Assim, este volume dos Anais Eletrônicos do ENUCOMP 2014 é composto por 11 capítulos: 6 relacionados aos mini-cursos trabalhados durante o encontro, e outros 5 formados pelos artigos científicos selecionados pelo Comitê de Programa para apresentação, mas não premiados.

Por último, gostaríamos de agradecer ao Corpo Editorial da RBCA (Revista Brasileira de Computação Aplicada) pela parceria na publicação dos 3 melhores artigos do nosso evento. O nosso muito obrigado também aos palestrantes, aos ministrantes de mini-cursos e aos membros da equipe de apoio e do comitê de programa, por acreditarem em nosso evento. O trabalho voluntário realizado por vocês foi de fundamental importância para o sucesso do ENUCOMP. Desejamos que o evento possa trazer frutos para o trabalho de todos.

Até a próxima!



Thiago Carvalho de Sousa
Coordenação Geral
ENUCOMP 2014

COMISSÃO ORGANIZADORA

Coordenação Geral:

Athânio de S. Silveira, Instituto Federal do Piauí (IFPI)
Rodrigo Augusto R. S. Baluz, Faculdade Piauiense (FMN)
Thiago C. de Sousa, Universidade Estadual do Piauí (UESPI)

Equipe de Apoio:

Anderson Passos Aragão, Faculdade Maurício de Nassau (FMN)
Antônio S. de Sousa, Instituto Federal do Piauí (IFPI)
Átila R. Lopes, Universidade Estadual do Piauí (UESPI)
Francisco das Chagas Rocha, Universidade Estadual do Piauí (UESPI)
Francisco Gerson Amorim de Meneses, Instituto Federal do Piauí (IFPI)
Lianna Mara Castro Duarte, Universidade Estadual do Piauí (UESPI)
Cornélia Janayna Pereira Passarinho, Universidade Federal do Piauí (UFPI)

Comitê de Programa:

Aldir Sousa, Universidade Estadual do Piauí (UESPI)
André Fujita, Universidade de São Paulo (IME-USP)
André Macêdo, Universidade Federal do Piauí (UFPI)
Antonio Kantek. Google
Aryldo Russo Júnior, CERTIFER
Atila Lopes, Universidade Estadual do Piauí (UESPI)
Carlos Giovanni Nunes, Universidade Estadual do Piauí (UESPI)
Celina Takemura, Embrapa
Christian Paz-Trillo, SAP
Claudia Melo, Thought Works
Constantino Dias, Universidade Estadual do Piauí (UESPI)
David Pereira, Banco Central do Brasil
Eduardo Leal, Universidade Federal do Pará (UFPA)
Eduardo Takeo Ueda, Centro Universitário SENAC-SP
Esdras Bispo Júnior, Universidade Federal de Goiás (UFG)
Erick Passos, Instituto Federal do Piauí (IFPI)
Eyder Rios, Universidade Estadual do Piauí (UESPI)
Fábio Gomes, Instituto Federal do Piauí (IFPI)
Fábio Kepler, Universidade Federal do Pampa (UNIPAMPA)
Fábio Siqueira, Universidade de São Paulo (POLI-USP)
Flávio Barros, Faculdade Maurício de Nassau (FMN)
Flávio Coutinho, Universidade de São Paulo (EACH-USP)
Francisco Rocha, Universidade Estadual do Piauí (UESPI)
Harilton Araújo, Centro de Ensino Unificado de Teresina (CEUT)
Haniel Barbosa, Universidade Federal do Rio Grande do Norte (UFRN)
Hermes Castelo Branco, Universidade Estadual do Piauí (UESPI)
Jaclason Machado, Universidade Federal do Piauí (UFPI)

Comitê de Programa (cont.):

Janayna Passarinho, Universidade Federal do Piauí (UFPI)
Jesus Mena, Universidade Federal do ABC (UFABC)
José Bringel Filho, Universidade Estadual do Piauí (UESPI)
Karina Valdivia, Universidade de São Paulo (EACH-USP)
Liliam Barroso, Universidade Estadual do Piauí (UESPI)
Márcio Monteiro, IBM
Marcos Couto, Oracle
Marcus Vinícius Lemos, Universidade Estadual do Piauí (UESPI)
Mehran Misaghi, Sociedade Educacional de Santa Catarina (SOCIESC)
Nécio Veras, Instituto Federal do Ceará (IFCE)
Paulo Silveira, Caelum
Pedro de Alcântara Neto, Universidade Federal do Piauí (UFPI)
Raphael Cobé, Instituto Federal do Rio Grande do Norte (IFRN)
Raimundo Barreto, Universidade Federal do Amazonas (UFAM)
Régis Magalhães, Universidade Federal do Ceará (UFC)
Ricardo Andrade, Universidade Federal do Piauí (UFPI)
Ricardo Sekeff, Faculdade das Atividades Empresariais (FAETE)
Rodrigo Baluz, Faculdade Maurício de Nassau (FMN)
Rodrigo Veras, Universidade Federal do Piauí (UFPI)
Seiji Isotani, Universidade de São Paulo (ICMC-USP)
Sérgio Barros, Universidade Estadual do Piauí (UESPI)
Thiago Carvalho de Sousa, Universidade Estadual do Piauí (UESPI)
Vladimir Rocha, Infomobile

SUMÁRIO

Mini-Cursos

Artigos

Aplicações para Interface Homem-Máquina baseadas em Automação com Arduino

Flávio Alves dos Santos [1]
Matheus Pereira Barros [1]

Resumo: A ascensão da tecnologia na manipulação de informações está muito acentuada no tocante à praticidade e execução de tarefas cotidianas. Diversas aplicações já utilizam tecnologia para seu provimento, como por exemplo tecnologia assistiva auxiliando a pessoas que necessitem deste a realizarem suas tarefas, a própria indústria do entretenimento, da mídia, da saúde e várias outras. Assim, um profissional atualizado na área abre diversas possibilidades de empregabilidade e até mesmo de inovação e alavancamento da reputação do estado. Com base nestas premissas, é que se deu o desenvolvimento de um minicurso que objetiva a capacitação de mão de obra para o desenvolvimento de tecnologia barata, mas com grande aplicabilidade.

***Palavras-chave*:** Arduino. Automação. Circuito. IHM.

Abstract: *The rise of technology in information handling is very sharp with regard to practicality and execution of daily tasks. Several applications already use technology to its provision, such as assistive technology helping people who need this to perform its tasks, the entertainment industry itself, the media, health and many others. Thus, a professional upgraded opens many possibilities in the area of employability and also to innovation and leveraging the state's reputation. Based on these assumptions, is that given the development of a short course which aims to train manpower for the development of cheap technology, but with wide applicability.*

***Keywords*:** *Arduino. Automation. Circuits. IHM*

1 Instrodução ao Arduino

Ardiuno é uma placa e uma linguagem de programação de baixo custo para prototipação de ciruitos eletro-eletrônicos. Ele, em conjunto com outros equipamentos, é capaz de realizar uma infinidade de tarefas mais complexas, como por exemplo o acionamento de motores, o processamento de sinais eletrônicos de sensores, o monitoramento de uma central de servidores, entre vários outros.

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments. [1]

O hardware que compõe a placa do Arduino é muito simples. Em suma, um microprocessador ATMega em conjunto com várias portas de entrada e saída de dados (I/O) que podem ser analógicas, ou digitais. Arduino é, na verdade, um conjunto de equipamentos para a realização de um único trabalho.

Seu ambiente de programação, o Arduino IDE (Integrated Development Environment - Ambiente de Programação Integrado) foi desenvolvido em Java, mas a linguagem que o seu compilador utiliza é baseada em Processing, também sendo possível a sua programação em C. Este IDE provê todas as funções necessárias para programa a placa. Funções como gravação de bootloader, verificação de código, tradução para código intermediário, upload de código binário, entre várias outras. A Figura 2 mostra a janela principal do Arduino IDE.

Há várias versões de placas Arduino. A mais comum de ser encontrada e utilizada é a versão UNO. Esta possui um microcontrolador de 16MHz de clock, uma memória flash de 32kB (está é compartilhada para o armazenamento do sketch gravado e a memória utilizada para a execução de tarefas), 2kB de SRAM, 1kB de EEPROM

[1]Pesquisador no Laboratório de Robótica, Automação e Sistemas Inteligentes - LABIRAS
{flavio@labiras.cc
matheuspereirabarros@gmail.com}

e suporta tensões de alimentação entre 6 e 18V (Volts). De forma segura, aconselha-se a alimentação deste com tensões entre 7 e 12V, devido às percas no seu regulador de tensão e a sobrecarga do mesmo. Ainda na versão UNO, temos 14 portas de entrada e saída digitais, destas, duas portas são exclusivas para integração com outros dispositivos de comunicação, quando estes são requeridos, exceto este caso, podem ser utilizadas normalmente. Possui 6 portas analógicas integradas a um conversor analógico-digital de 10 bits de resolução. A própria placa Arduino é capaz de fornecer energia contínua para alimentação ou acionamento de outros circuitos. Assim, fornece uma tensão de 5V a uma corrente em torno dos 300mA (miliamperes) nas portas de alimentação. Já nas portas digitais a corrente é reduzida aos 150mA devido a esta ser fornecida pelo microprocessador e não direto do regulador de tensão. O esquemático de referência da Arduino UNO pode ser encontrado no seguinte endereço: http://arduino.cc/en/uploads/Main/arduino-uno-schematic.pdf.

Podemos visualizar a placa Arduino UNO na Figura 1.

Figura 1: Arduino UNO. Fonte: http://store.arduino.cc/

1.1 Requisitos e Instalação

Para instalar Arduino IDE, precisaremos ter instalado no computador o Java Runtime Environment (JRE). Recomendamos o uso do Oracle Java, disponível no link http://java.com/download/. Escolha a opção de acordo com o seu Sistema Operacional e o instale. Após as configurações do ambiente Java, o Arduino IDE já poderá ser aberto para uso.

Tenha a certeza de que você seu usuário é um administrador do sistema, pois você precisará instalar os programas e a placa durante a primeira conexão com o computador. Neste último procedimento, no caso do Windows, pode não ser reconhecida automaticamente e o driver para a placa não ser instalado, para isso, você deverá navegar até seu Gerenciador de Dispositivos e fazê-lo manualmente. Em caso de dúvidas, consulte o manual, ou suporte de seu Sistema Operacional.

Visite o Arduino Playground para verificar projetos, exemplos, códigos fonte, entre outros. Arduino Playground pe um trabalho contínuo. Nele é possível participar ativamente da comunidade Arduino dando contribuições e recebendo ajuda de outros membros [1].

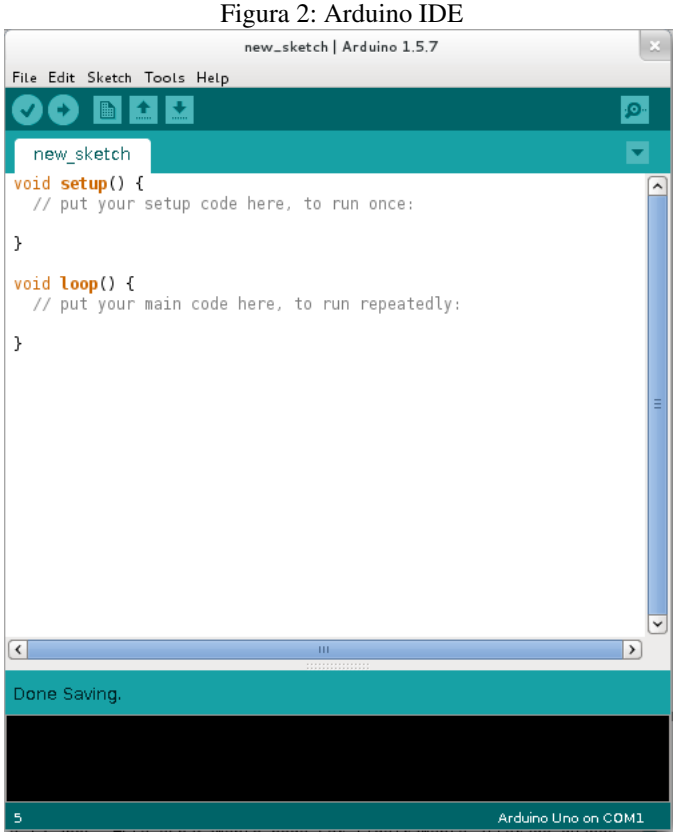
Obs. 1: Siga as instruções de instalação e configuração do Java de acordo com o manual do fabricante.

Obs. 2: Verifique a versão de seu Sistema Operacional e a arquitetura do mesmo. Programas com arquitetura de 64bit não são compatíveis com 32bit, mas o contrário sim, porém não obtém o máximo de desempenho ao ser executado.

Obs. 3: No site oficial há duas versões do Arduino IDE. A primeira é a última versão esstável dele. Esta é a mais recomendada para usuários novatos. A segunda é a última versão de testes, chamada de 'beta'. Esta última é a versão com novas funcionalidades, que pode ter sido lançada, mas sem testes necessários para

validá-la como versão oficial, podendo assim, conter alguns erros de execução. Mesmo sendo versão de testes, os autores recomendam o seu uso, tanto pelas novas funcionalidades, pelas facilidades implementadas a cada nova versão, quanto pela oportunidade de ajuda à comunidade, indicando novos erros que venham a surgir, ajudando no desenvolvimento do IDE.

Figura 2: Arduino IDE



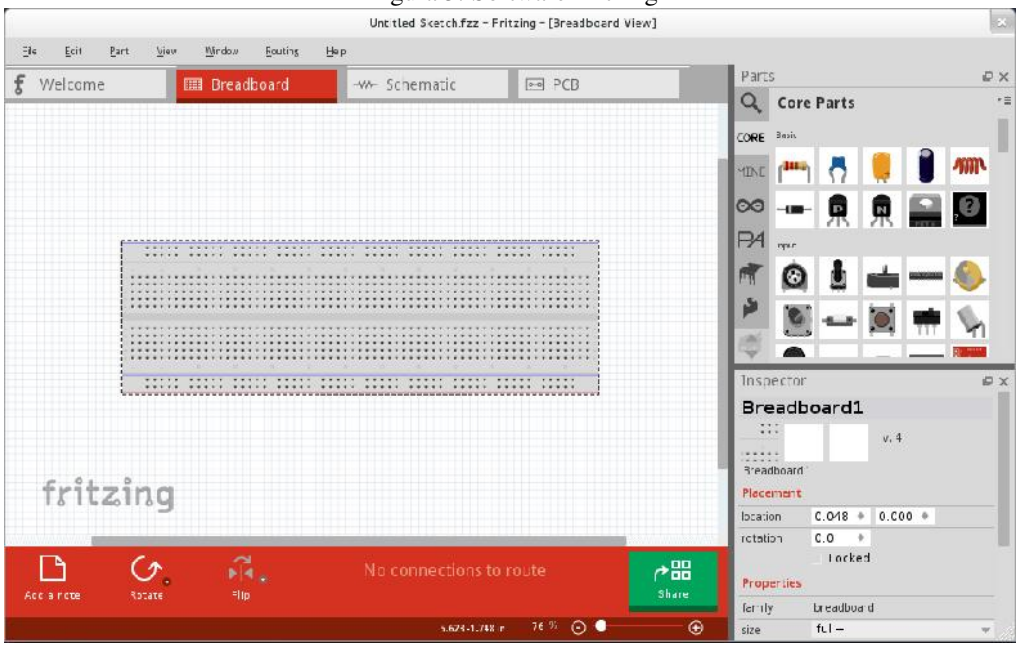
2 Fritzing

Fritzing é um projeto Open Source para documentação e projeto de hardware. Proporciona um ambiente intuitivo e simples para o usuário similar a um ambiente real de prototipação. Possui três visões, são eles: Protoboard, Esquemático e PCB. A primeira visão simula a prototipação em uma protoboard, onde o esquema de ligação é similar à desta. A segunda é a montagem através do esquemático dos componentes; é mais utilizado por profissionais de eletrônica. A terceira é a visão capaz de organizarmos um esquema para exportação de uma placa de circuito impresso (PCI, ou PCB - Printed Circuit Board). Nesta última, podemos organizar melhor os componentes montados na visão Protoboard e/ou Esquemático e exportarmos uma placa personalizada, como por exemplo, uma nova shield para Arduino, uma placa de um amplificador de áudio, uma placa de um transmissor de FM entre vários outros.

Os requisitos para a instalação e uso do Fritzing são os mesmos do Arduino IDE. Precisamos apenas do JRE instalado para sua execução. Devemos lembrar de obter sempre a versão compatível com o nosso Sistema Operacional.

Podemos visualizar na Figura 3 a janela do Fritzing e suas funcionalidades. Mais detalhes, visite http://fritzing.org/. Neste site é possível fazer o download do software, exemplos, projetos e novos componentes.

Figura 3: Software Fritzing



3 Aplicações

Como citado anteriormente, Arduino facilita a prototipação de circuitos de controle, circuitos de sensoreamento, circuitos de acionamento, entre vários outros. Podemos destacar a sua versatilidade na integração com diversos tipos de equipamentos. A seguir teremos alguns exemplos destas aplicações com Arduino.

3.1 Sensores

Um sensor é um dispositivo que, a partir da energia gerada no meio onde se é medido, um sinal de saída é transmitido para a função que o avalia. [2]. Sensores são qualquer equipamento que, ao se aplicar um estímulo, ele gera um sinal que pode ser analisado e utilizado por um dispositivo de processamento e produzir um resultado.

Podemos citar como exemplo de sensores, LDR (Light Dependent Resistor - Resistor Dependente de Luz) que, na verdade é um resistor variável que, aplicando-se níveis de luz o mesmo, ele produza resultados diferentes para estas condições. Sensores de presença consistem em enviar um sinal (sonoro, visual, ou mesmo um feixe de luz) e conseguir distinguir estados diferentes. Ex.: HC-SR04 que é um sensor de distância por ultrassom. O sensor envia um pulso sonoro ao meio, e calcula o tempo que o mesmo sinal retorna ao sensor. Assim, pode-se saber a distância emtre o sensor e o obstáculo.

3.2 Atuadores

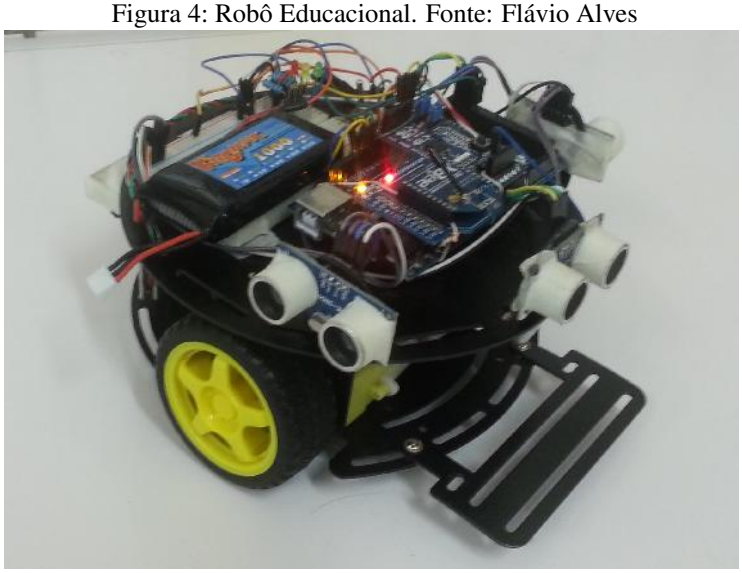
Atuadores são quaisquer dispositivos capazes de modificar o meio em que são submetidos. Como exemplo destes podemos citar os Relés, que são capazes de acionar uma corrente de alta potência a partir de uma corrente de baixa potência. Podemos citar também motores que realizam movimentos de acordo com a programação no microcontrolador.

3.3 Automação e Robótica

As aplicações mais comuns são na área de robótica e automação. Podemos citar como exemplo o desenvolvimento de um protótipo de um robô para o ensino de matemática e física para o Ensino Médio das escolas

públicas. A Figura 4 mostra a foto do robô finalizado. Já na Figura XXXX podemos visualizar o esquema de montagem do protótipo do mesmo.

Figura 4: Robô Educacional. Fonte: Flávio Alves



3.4 Saúde e Tecnologia Assistiva

A chamada tecnologia assistiva (ou inclusiva) tem avançado e vem permitindo a deficientes físicos, visuais e locomotores novas formas de aprendizado, que garantem mais possibilidades para se inserirem na sociedade [3]. O conceito de assistividade não vem apenas da necessidade de suprir a falta de alguma função natual do corpo por funções artificiais como próteses robóticas, exoesqueletos, ou mesmo luvas tecnológicas, mas da necessidade de se auxiliar ea execução de atividades diárias, normalmente simples, através da tecnologia.

Podemos citar como exemplo de tecnologia assistiva uma luva para deficientes visuais, uma cadeira de rodas eletrônica inteligente, um aplicativo que indique o horário que um coquetel de medicamentos deve ser servido, etc.

A produção de medicamentos médicos, normalmente, se utiliza de sensores captadores de sinais e um microcontrolador acomodador destes sinais eletrõnicos. O difirencial destes equipamentos é como estes sinais são tratados e os dados que eles podem produzir a partir de uma massa de dados extensa. O processamento dos dados depende da aplicação que este deve realizar. Pro exemplo, um ultrassom que informa ao usuário, em tempo real o estado da área sensoreada, um eletrocardiograma portátil que se comunica com o paciente através de seu smartphone, entre várias outras aplicações. Depende, intrinsecamente, do projetista especialista definir e limitar o escopo do equipamento e as rotinas que este deverá executar.

3.5 Games

A utilização de hardware na área de entretenimento não é nova. Circuitos eletrônicos de controle são utilizados desde o período dos jogos de azar eletrõnicos, casas de bingo eletrônicos, vídeo games, etc. O diferencial dos equipamentos mais modernos é a integração destes com os aplicações cada vez mais complexas.

A produção de controladores de jogos é largamente difundida. Telas sensíveis ao toque abrem infinitas possibilidades para a área. Jogos como 'Guitar Hero', 'BassHero', 'Karaokê', etc. se utilizam deste tipo de equipamento para proverem um maior nível de interação e entretenimento. Consoles como 'PlayStation', 'XBox', 'Nintendo Wii', entre outros, famosos por seu poder de processamento de jogos em alta resolução, também utilizam controladores de jogos integrados a ele.

As possibilidades são tantas que a cada nova aplicação surge um novo controlador de jogo. Um grande

exemplo são os controladores de 'E-Sports' (Esportes Eletrônicos) que, dependendo do segmento, requer uma nova forma de interação, consequentemente, um novo controlador.

4 Programando a placa Arduino

Assim como em várias linguagens de programação, devemos ter uma estrutura mínimo para a execução de uma tarefa, no Arduino também devemos ter esta estrutura. Precisamos definir as funções setup() e loop().

A função 'void setup()' é utilizada para a configuração do hardware do arduino para o uso pelo mesmo. Por exemplo, configuramos se iremos escrever, ou ler alguma informação em uma porta digital; ou se precisarmos enviar alguma informação para algum outro dispositivo, devemos informar a taxa de transferência; inicialização de valores de variáveis também podem ser feitas nesta área, entre outros.

A função 'void loop()' será nosso Sistema Operacional no Arduino. Nesta função deveremos descrever todo o comportamento do projeto. Nesta função devemos programar todas os procedimentos que a Arduino deverá fazer para executar suas ações, como cálculos, envio de mensagens para outros dispositivos, leitura dos valores de sensores, acionamento de circuitos, dentre outros.

Perceba que nenhuma das funções precisa ser inicializada, ou mesmo feito uma chamada a ela. Acontece que o próprio bootloader do Arduino já se encarrega disto. Ao inicializar, o bootloader faz a chamada à função setup() e logo em seguida faz chamadas seguidas da função loop(). Resumindo, a função setup() é executada apenas uma vez, no início dos procedimentos. Já a função loop() é executada sequencialmente. Sempre que os procedimentos desta função são finalizados, é feita uma nova chamada para a função. Por isso dizemos que a função loop() é o Sistema Operacional. É nela que devem estar as instruções para a Arduino realizar suas tarefas.

Código mínimo do Arduino:

```
1. void setup(){}
2.
3. void loop(){}
```

Exemplo: Blink - Piscando um LED com Arduino:

```
1.  void setup(){
2.    pinMode(13, OUTPUT);
3.  }
4.
5.  void loop(){
6.    digitalWrite(13, HIGH);
7.    delay(1000);
8.    digitalWrite(13, LOW);
9.    delay(1000);
10. }
```

Discutindo o exemplo acima, temos o seguinte comportamento: a cada segundo um LED, conectado à porta digital de número 13 da Arduino mudará de estado. Assim, ao se chamar a função loop(), o LED acende (devido ao parâmetro HIGH), em seguida, espera-se um segundo, desliga-se o LED (conectado à porta digital 13) e espera-se mais um segundo. Como a função loop() é chamada intermitentemente, o LED acende e apaga intermitentemente, simulando a ação de um pisca-pisca.

```
1.  const int buttonPin = 2;
2.  const int ledPin =  13;
3.
4.  int buttonState = 0;
5.
```

```
6.  void setup() {
7.    pinMode(ledPin, OUTPUT);
8.    pinMode(buttonPin, INPUT);
9.  }
10.
11. void loop() {
12.   buttonState = digitalRead(buttonPin);
13.
14.   if (buttonState == HIGH) {
15.     digitalWrite(ledPin, HIGH);
16.   } else {
17.     digitalWrite(ledPin, LOW);
18.   }
19. }
```

O código acima já se utiliza de dois componentes. Um botão (*push button*) e um LED. Ao se pressionar o botão, o LED é ligado. Quando o botão for liberado, o LED será desligado.

4.1 Características da Linguagem

O Arduino provê três principais partes: estrutura, valores (variáveis e constantes) e funções [1].

- Estrutura
  - setup()
  - loop()

  Estruturas de Controle
  - if
  - if...else
  - for
  - switch case
  - while
  - do... while
  - break
  - continue
  - return
  - goto
- Valores

  Constants
  - HIGH | LOW
  - INPUT | OUTPUT
  - true | false

  Data Types
  - void
  - boolean
  - char
  - unsigned char
  - byte
  - int
  - unsigned int
  - word
  - long
  - unsigned long
  - short
  - float

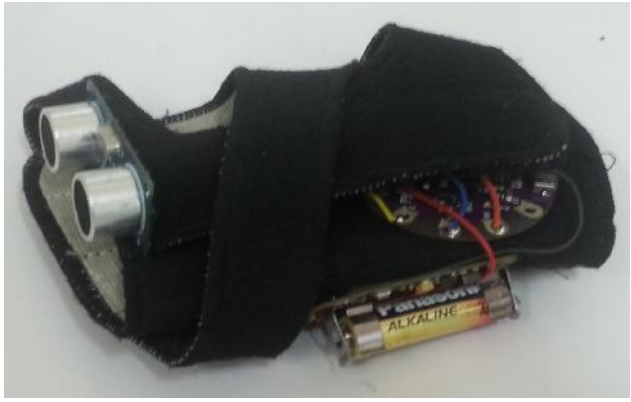
  - double
  - string - char array
  - String - object
  - array

- Funções
  - pinMode()
  - digitalWrite()
  - digitalRead()
  - analogWrite()
  - analogRead()
  - delay()
  - millis()
  - micros()

5 Trabalhos Relacionados

Dentre vários trabalhos desenvolvidors no LABIRAS, citamos, principalmente três deles. O primeiro é o projeto protótipo de uma luva para deficientes físicos, a Luva Ultrassônica. Este projeto, de tecnologia assistiva, tem o objetivo de auxiliar a deficientes visuais cegos, ou com baixa visão a se locomover de forma mais fácil em um ambiente fechado (*indoor*). O projeto consistem, basicamente, em um sensor detector de obstáculos, uma placa Arduino e um atuador, no caso, um motor vibracall. O Arduino obtem os dados do sensor detector de obstáculos, quando estes dados obedecem aos parãmetros definidos na codificação como reconhecimento de um obstáculo, o Arduino aciona o motor que faz com que o deficiente possa sentir uma vibração na luva, indicando que há um obstáculo à sua frente. Uma foto da mesma pode ser visualizada na Figura 5.

Figura 5: Luva Ultraassônica. Fonte: LABIRAS



O segundo projeto que destacamos é o projeto de uma bengala eletrônica, também na área de tecnologia assistiva objetiva o auxilíar a um cego na sua locomoção em ambientes abertos. Normalmente, um cego se utiliza de uma bengala comum sem tecnologia. A inovação existente nesta bengala é que adiciona funcionalidades à bengala comum. Através de um detector de obstáculos, o cego poderá ser informado de obstáculos que uma bengala comum não detectaria, como por exemplo, um orelhão, uma marquize baixa, ou mesmo um desnível de ambiente. A bengala comum faz a detecção de obstáculos e desníveis. O que propomos é uma antecipação desta detecção, evitanto com que o usuário venha a se machucar em acidentes de colisão com estes obstáculos. Observe na Figura 6 a foto da Bengala Eletrõnica.

O terceiro projeto relacionado que destacamos é uma bateria eletronica de baixo custo para o ensino de música. Este projeto da área de entretenimento e automação tem por objetivo diminuir o custo de um equipamento musical e torná-lo acessível para que mais pessoas possam ter a oportunidade de aprender a utilizar um instrumento como esse. Este consiste em uma placa acomodadora de sinal (Arduino) e sensores que captam os impactos nos

Figura 6: Bengala Eletrônica. Fonte: LABIRAS

pads. Ao se captar um impacto em um, ou mais pads, o Arduino envia ao computador conectado um pacote MIDI para a reprodução do mesmo através do computador. Neste caso, além do baixo custo, é também objetivado a facilitação na configuração da bateria musical no tocante à microfonação e configuração de ambiente surround. Podemos visualizar a foto do protótipo na Figura 7.

Figura 7: Bateria Eletrônica. Fonte: LABIRAS

6 Conclusões

Com tantas possibilidades de alteração do ambiente, as aplicações de Interação Homem-Máquina se encaminham à realidade da Internet das Coisas (IoT - Internet of Things), onde cada objeto que interagimos poderão se comunicar para prover melhores serviços e realizar tarefas independentemente da interação humana.

A ciência e a tecnologia se encaminham, a passos largos, a uma realidade bem próxima onde equipamentos eletrônicos serão a razão da saúde de grande parte da população, de entretenimento, de locomoção, etc. Cabe a cada usuário/projetista, saber utilizá-lo com consciência para que pessoas humanas não se tornem dependentes destes, por suas falhas, insegurança, entre outros.

Agradecimentos

Os autores agradecem a comissão organizadora do Encontro Unificado de Computação de Parnaíba pela oportunidade de difundir os conhecimentos aqui demonstrados e descritos.

A todos os membros do LABIRAS - Laboratório de Robótica, Automação e Sistemas Inteligentes do Instituto Federal do Piauí - IFPI, *Campus* Teresina Central pela ajuda e acolhida aos autores. De certa forma, todos contribuem positivamente para o crescimento e difusão da ciência no Estado do Piauí.

A todos os interessados no tema, pois, com isso, é possível haver uma manutenção e elevação do conhecimento. A produção científica e técnica do estado depende disso.

Referências

[1] ARDUINO. *Arduino - Home*. 2014. Disponível em: <http://arduino.cc/>.

[2] ARENY, R. P. *Sensores y acondicionadores de señal*. [S.l.]: Marcombo, 2003.

[3] MOREIRA, A. L. S.; VASCONCELOS, F. N. et al. Uso de robótica assistiva no auxílio de pessoas com deficiências visuais. In: *V CONNEPI-2010*. [S.l.: s.n.], 2010.

Deixe sua imaginação fluir: Desenvolvimento de Jogos para *Android* com o *Framework Cocos* 2D

George Max P. Souza[1], Ádrian C. Oliveira[1]

[1]Universidade Estadual do Piauí (UESPI)
Caixa Postal 64.200-000 – Parnaíba – PI – Brasil

{georgemaxphb, adriancoliviera}@gmail.com

Abstract. *The proposed work aims to explain a growth area as the development of games for Android, and make a brief presentation of some steps to be followed by developers in the preparation of a draft of a play, and show how the scenario is this current market and briefly describe a framework used for building 2D games, which has distributions for various platforms, among them the Android platform, which is the object of focus in this work.*

Resumo. *O proposto trabalho tem como objetivo explanar uma área em crescimento como a de desenvolvimento de jogos para Android, bem como fazer uma breve apresentação de alguns passos a serem seguidos por desenvolvedores na elaboração de um projeto de um jogo, além de mostrar como está o cenário atual deste mercado e descrever de forma breve um framework utilizado para construção de jogos 2D, o qual possui distribuições para diversas plataformas, dentre elas a plataforma Android, que é objeto de foco neste trabalho.*

1. Mercado de Jogos Móveis

No cenário competitivo do desenvolvimento de jogos, um novo ambiente vem se destacando e adquirindo adeptos rapidamente, são os dispositivos móveis, como por exemplo, celulares, *smartphones*, *tablets* e uma infinidade de outros aparelhos. Empresas de jogos e até mesmo desenvolvedores independentes (*Indie Games*) estão criando novas oportunidades de negócios.

Segundo Korjenioski (2011), do período de 2009 para 2010, este segmento teve um crescimento de 5% para 8%, avançando na fatia dos jogos para consoles e portáteis de empresas já consagradas, como por exemplo, *Nintendo*, *Sony* e *Microsoft*. Uma pesquisa mais recente, realizada pelo Serviço Brasileiro de Apoio às Micro e Pequenas Empresas (SEBRAE), revela que o Brasil é o quarto maior mercado do mundo dentro do setor de jogos digitais, com 35 milhões de usuários. A pesquisa se encerra abordando que o mercado de *games* movimentou R$ 5.3 bilhões em 2012, com crescimento de 32% em relação a 2011 e ainda com perspectivas positivas de crescimento para novos empreendedores até 2016.

É possível perceber que no Brasil, além de recente, o mercado de jogos está cada vez mais consolidado e aberto a novas ideias, tanto para desenvolvedores quanto para jogadores, até empresas estrangeiras apostam na tradução de seus *games* para o português logo após o lançamento.

A figura 1 a seguir mostra a maior feira de *games* da América Latina, a Brasil *Game Show* 2014, que aconteceu em São Paulo no mês de Outubro.

Figura 1. Brasil *Game Show* 2014.

2. Jogos para *Android*

O avanço tecnológico possibilitou a crescente evolução dos dispositivos móveis, os quais adquiriram, assim como os computadores, um alto poder de processamento, contando com a vantagem de oferecerem maior confiabilidade aos usuários e melhores experiências, além da possibilidade da criação de aplicações mais robustas (Silva, 2011).

O surgimento de plataformas que rodam em tais dispositivos também contribuiu para que os mesmos se popularizassem, sendo uma destas plataformas o *Android*, que possui maior popularidade entre as existentes por ser *software* livre, de código aberto e ainda permitir que desenvolvedores utilizem todos os recursos disponíveis nos aparelhos (Medeiros, 2013).

A facilidade da utilização de recursos dos dispositivos pela plataforma *Android* contribuiu para o crescimento do desenvolvimento de jogos para a tal, pois pode-se utilizar todos estes recursos para enriquecer os jogos e torná-los mais atrativos, onde Silva (2011) cita como exemplos de recursos: a câmera, o GPS, 3G, Tela Sensível ao Toque, Banco de Dados Nativo, entre outros. Jogos como *Fruit Ninja* (figura 2) e *Need for Speed Shift* (figura 3), ambos lançados em 2013, são exemplos, pois utilizam recursos como a Tela Sensível ao Toque (para simular uma espada utilizada para cortar as frutas) e o acelerômetro (para simular um volante) (Medeiros, 2013).

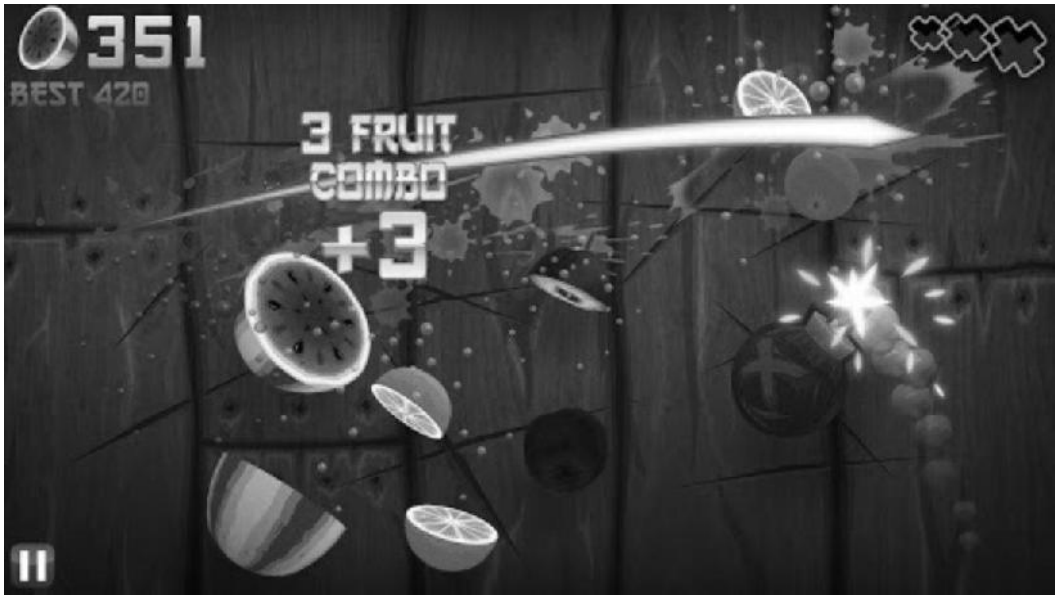


Figura 2. Jogo *Fruit Ninja*.

Figura 3. Jogo *Need for Speed Shift*.

3. O *Framework Cocos* 2D

O desenvolvimento de jogos é uma tarefa que oferece vários desafios a serem superados pelas equipes de desenvolvimento, devido à presença de características complexas como a implementação de sua lógica, dentre outros aspectos como interação com o usuário, manipulação de imagens e sons, simulação de eventos físicos e até mesmo de sua própria jogabilidade.

O uso de *frameworks* é cada vez mais adotado pelas empresas ou até mesmo por desenvolvedores independentes, tendo em vista o aumento da produtividade e a

facilidade que estes oferecem na utilização de recursos e por trazerem consigo uma grande quantidade de ferramentas (funcionalidades) prontas.

O *framework Cocos* 2D é um motor gratuito, de código aberto e possui versões em diversas outras plataformas como computadores pessoais e outras para dispositivos móveis como *Android*, *Windows Phone* e *Bada*. Ele é uma solução focada no desenvolvimento de jogos 2D com uma filosofia menos técnica, pois não possui ferramentas que diminuam a necessidade de programação direta usando uma linguagem, mas evita que programadores tenham contato direto com funcionalidades de baixo nível (Benin, 2012).

4. Etapas de Desenvolvimento de um Jogo

Um jogo requer uma série de etapas durante o seu ciclo de vida, passando pela fase de criação do protótipo do jogo, telas do jogo, inimigos, pontuação, efeitos sonoros, dentre outras que serão descritas com mais detalhes logo em seguida.

4.1. Protótipo do Jogo

Nessa fase é que os membros da equipe de desenvolvimento definem todas as características que serão implementadas no jogo, bem como o *design* de telas e o contexto ao qual o mesmo está inserido. Dessa forma é possível comunicar para todos os membros os objetivos do *game*, fazendo com que possíveis detalhes complexos sejam evitados não impedindo seu andamento e/ou conclusão.

Na figura 4 abaixo é possível visualizar o protótipo inicial do jogo *Plants vs. Zumbies*, onde se é desenhado quadro por quadro da interação entre jogadores e inimigos.

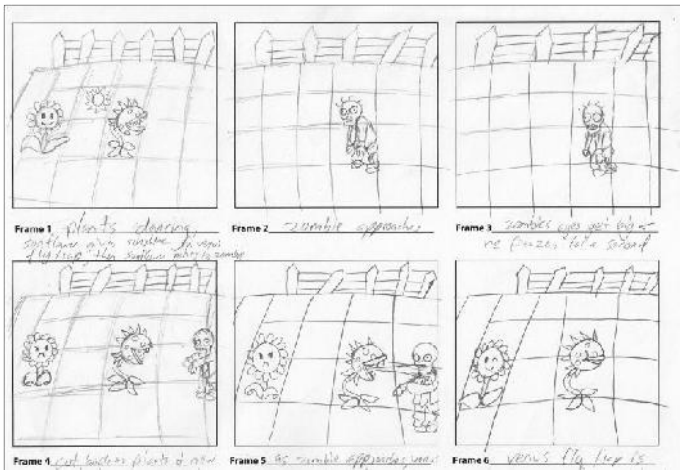


Figura 4. Protótipo quadro a quadro do jogo *Plants vs. Zumbies*.

4.2. Telas do Jogo

Todas as telas planejadas na etapa anterior de prototipação serão agora desenvolvidas. Conceitos de background (plano de fundo), animações e etc., são fundamentais para esse estágio, além da comunicação direta entre os *designers* (profissional responsável pela estética do jogo) e programadores.

4.3. Inimigos

Com as telas definidas e desenvolvidas, passa-se agora para a implementação da lógica de negócios aplicada aos inimigos dentro do jogo. É possível, isso depende da complexidade do game, que sejam utilizados técnicas de IA (Inteligência Artificial), dando mais realismo e tornando o jogo mais competitivo e divertido para os jogadores. Ainda nessa fase são definidos e executados alguns testes para a validação dos inimigos criados, para que seja possível passar para a próxima etapa.

4.4. Pontuação

Para que os jogadores do jogo sintam-se motivados a jogar, além de uma boa história, eles precisam ganhar pontos por ações efetuadas durante os jogos, por isso é importante ter o planejamento do que valerá cada ação efetuada, tanto no acréscimo quanto na retirada de pontos. Nesse contexto poderão existir, pontos de vida, pontos extras que ajudarão ou fornecerão aos jogadores algum prêmio e ainda, em alguns casos, a venda de pontos, situação muito comum e lucrativa no cenário atual dos jogos para dispositivos móveis e que vem ganhando cada vez mais adeptos.

Logo abaixo, na figura 3, é mostrada a tela do jogo *Pou* para *Android*, onde se tem a possibilidade de obter diversos itens do jogo através do cartão de crédito.

Figura 4. Compra de itens no jogo *Pou* para *Android*.

4.5. Efeitos Sonoros

Para passar mais realismo ao jogador, os jogos digitais podem oferecer efeitos sonoros diversos para transmitir indicações de situações dentro dos jogos, como por exemplo, vitória do personagem, captura de algum prêmio, tempo se esgotando e até mesmo a trilha sonora de fundo para passar adrenalina ou tranquilidade aos jogadores.

5. Publicando na *Google Play*

Google Play é uma ferramenta disponibilizada pela *Google*, onde programadores da plataforma *Android* podem publicar seus aplicativos e compartilhá-los com outros usuários da plataforma e de acordo com Keller (2012) , qualquer pessoa ou empresa que tenha uma conta de desenvolvedor pode distribuir aplicativos pelo mesmo (Medeiros, 2013).

Para publicar uma aplicação (seja ela um jogo ou não) são necessários alguns passos (Keller, 2012):

- Registrar-se no *Google Play*, concordar com o termo disponível no *site* e pagar uma taxa de cadastro.
- Enviar o aplicativo com extensão .apk (*Android Package File*), cujo tamanho máximo deve ser de 50 *Megabytes*.
- Enviar pelo menos duas imagens de captura de tela do aplicativo.
- Enviar um ícone em alta resolução do aplicativo.
- Publicar seu aplicativo.

6. Conclusão

Com base no que foi abordado nas seções anteriores, conclui-se que o mercado de jogos vem crescendo, principalmente para plataformas móveis, devido a estas estarem cada vez mais emergindo e tornando-se populares, nos levando a ter uma visão de que futuramente poderão ser substitutas dos computadores. Foi visto também que a construção de um jogo é complexa, mas que estes possuem etapas que quando seguidas dão ao desenvolvedor uma melhor visão do que deverá ser feito, além da abordagem sobre *frameworks*, que como dito, são cada vez mais utilizados por serem ferramentas que auxiliam no desenvolvimento de um projeto, diminuindo o tempo de construção e fornecendo inúmeras facilidades, inclusive para aqueles que não possuem um conhecimento aprofundado da plataforma a ser utilizada para o desenvolvimento.

7. Referências

Korjenioski, M. (2011) “Desenvolvimento de Jogos 2D com Android”, Curitiba.

Silva, S. S. and Nobrega, S. M. and Junior, A. F. L. J. (2011) “Labirinto do Rato: jogo educacional infantil para dispositivos móveis”, Aracaju.

Medeiros, L. R. (2013) “Desenvolvimento de um jogo para Android utilizando recursos de acelerômetro e Tela Sensível ao Toque”, Pato Branco.

Benin, M. R. and Zambiasi, S. P. (2012) “Proposta de uma Ferramenta Focada no Ensino do Desenvolvimento de Jogos Eletrônicos”, Santa Catarina.

Keller, M. A. (2012) “Jogo de pôquer texas hold'em para Android”, Curitiba.

Mineração de Opiniões: desafios e técnicas

Francisco A. Ricarte Neto[1], **Rafael T. Anchiêta**[1]

[1]Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Recife – PE – Brasil

{farn,rta2}@cin.ufpe.br

Abstract. *This paper presents an overview about Opinion Mining. It is presented some motivations that conducted several researchers to study this field, the definition of the term Opinion Mining, and also some of the main concepts of the area, like the granularity-levels of Opinion Mining systems and the phases to create a complete process. Next, it is showed few major approaches used in related works in Opinion Mining. Finally, it is pointed some challenges that presents difficulties to analyze opinions.*

Resumo. *Este artigo apresenta uma visão geral sobre a área de Mineração de Opiniões. É apresentado um pouco da motivação a qual levou diversos pesquisadores a estudar essa área, a definição do termo Mineração de Opiniões e alguns conceitos fundamentais como os níveis de granularadade de sistemas de Mineração de Opiniões, e as estapas para a realização de um processo completo. Em seguida, destacam-se as principais abordagens utilizadas para minerar opiniões relacionando a alguns trabalhos da área. Por fim, apontam-se alguns desafios que dificulta a análise de opiniões.*

1. Introdução

A Web 2.0 desencadeou uma explosão de serviços disponíveis na Internet, bem como um grande aumento de usuários na rede. Difundiu-se a criação de blogs, fóruns de discussão, sites de debates, redes sociais e sites de compras, favorecendo o aumento na interação entre os indivíduos.

Neste cenário, uma atividade que se tornou bastante frequente é a busca por opiniões sobre produtos e/ou serviços. De forma geral, todos os usuários recorrem a sites de opiniões (*reviews*), fóruns de discussão ou até mesmo às redes sociais em busca da experiência de outras pessoas antes de tomar suas decisões. Uma pesquisa de opinião realizada nos EUA revelou que entre 73% a 87% das pessoas que buscam opiniões na Web se disseram fortemente influenciadas pelas *reviews*[1]. Os entrevistados afirmaram ainda que estariam dispostos a pagar até 99% a mais por um serviço avaliado como "excelente" em relação a outro avaliado como "bom".

Os dados acima confirmam o fato de que a Internet se transformou em um poderosíssimo meio de transmissão de opiniões sobre produtos e serviços. Contudo, apesar de ser fácil encontrar opiniões na grande rede, a sua análise manual não é um processo trivial. Isso se deve a alguns fatores principais como: a abundância de informações disponíveis, a dificuldade de se interpretar linguagem natural, que é imprecisa e ambígua

[1]http://www.comscore.com/press/release.asp?press=1928

por natureza e as opiniões falsas, ou propositadamente tendenciosas sobre algum produto [Jindal and Liu 2007]. Logo, para se obter uma posição geral "confiável" dos usuários acerca de algum produto, é necessário buscar e analisar uma quantidade significativa de opiniões na Web.

Nesse contexto, a Mineração de Opiniões também conhecida como Análise de Sentimento (AS) vem facilitar a vida dos usuários que buscam a análise de opiniões de forma automática na Web [Liu 2012]. A Análise de Sentimentos, dentre outras tarefas, preocupa-se em classificar opiniões expressas em textos a respeito de um determinado objeto (produto, serviço, instituição ou pessoa) como positivas (e.g., o celular é incrível $(+)$), ou negativas (e.g., o filme é muito ruim $(-)$). Em geral, a classificação é feita com base nos adjetivos (e.g., bom, ruim, etc.), advérbios (e.g., rapidamente, bem, etc.), substantivos (e.g., amigo, inimigo, etc.) e verbos (e.g., odiar, amar, etc.) encontrados no texto. Por fim, a AS busca disponibilizar o resultado da análise para o usuário final de forma simples e clara. A AS procura realizar, de forma rápida e automática, a análise das opiniões disponíveis acerca de um produto ou serviço e retornar para o usuário a opinião geral sobre o objeto de seu interesse.

Diante do que foi apresentado, o objetivo deste trabalho é apresentar os principais conceitos, desafios e técnicas relacionados à Mineração de Opiniões. Dessa forma, o restante do artigo está organizado da seguinte maneira: Seção 2 apresenta os níveis de granularidade e as etapas de um processo de AS, Seção 3 discute sobre as técnicas mais utilizadas na construção de sistemas de AS, Seção 4 ilustra alguns dos principais desafios encontrados na construção de sistemas para minerar opiniões, e por fim, Seção 5 apresenta as considerações finais deste trabalho.

2. Mineração de Opiniões

A Mineração de Opiniões tem por objetivo identificar o sentimento expresso em *textos opinativos* (i.e., textos que contém opiniões sobre um tópico/objeto de interesse) [Liu 2012]. Dessa forma, define-se que *textos opinativos* são aqueles que possuem sentenças ou expressões *subjetivas*. Já os textos que não possuem expressões *subjetivas* são conhecidos como textos *objetivos* ou *factuais*. As opiniões podem ser classificadas entre positivas, negativas e neutras indicando assim a *polaridade* do texto (a polaridade neutra ocorre quando o texto traz opiniões negativas e positivas na mesma proporção). Em geral, a polaridade de um texto é expressa por palavras opinativas (adjetivos – bom, ruim; advérbios – bem, rapidamente; e alguns substantivos – amigo, etc).

O restante da seção apresenta alguns conceitos fundamentais sobre os níveis de granularidade da Análise de Sentimento, além das etapas para a realização de um processo completo de Mineração de Opiniões.

2.1. Níveis da Mineração de Opiniões

Segundo Liu (2012), a Mineração de Opiniões pode ser realizada em três níveis de granularidade, são eles: nível do documento, nível da sentença e nível do atributo [Liu 2012].

Mineração de Opiniões no nível do Documento busca determinar se cada documento expressa uma opinião geral positiva, negativa ou neutra a respeito do objeto sob análise. Note que, a polaridade de um documento opinativo é considerada neutra quando ele traz a mesma quantidade de avaliações positivas e negativas sobre o objeto em análise.

Como exemplo de trabalho neste nível de classificação, cita-se [Pang et al. 2002] que utilizaram algoritmos de aprendizagem de máquina para classificar *reviews* de filmes entre positivos ou negativos. Outros exemplos deste nível de AS são: [Sebastiani 2002] e [Dave et al. 2003].

Mineração de Opiniões no nível da Sentença geralmente se divide em duas etapas. Primeiramente, busca-se identificar se cada sentença do texto é subjetiva ou objetiva. Em seguida, procura-se determinar se as sentenças subjetivas expressam opiniões positivas, negativas ou neutras. Alguns trabalhos de AS nesse nível de classificação: [Yu and Hatzivassiloglou 2003] e [Kim and Hovy 2004].

Exemplo 1: Comentário retirado e traduzido do debate Sony Ps3 vs. Nintendo Wii[2].
*"***(1)** *Os controles dos PSs são péssimos.* **(2)** *A maior parte deles quebram.* **(3)** *Os controles do Wii são os mais intuitívos que já existiram.* **(4)** *Como um jogador casual de games, eu amo o meu Wiimote e esta é a primeira "certeza" que já tive na vida. "*

Por fim, na **Mineração de Opinião no nível do Aspecto**, os atributos são analisados isoladamente. Esse tipo de classificação é realizado por aplicações que procuram um nível maior de refinamento na AS. Depois de determinar (manualmente ou automaticamente) os atributos dos objetos sob análise, é realizada a classificação das opiniões sobre cada um dos atributos mencionados. No exemplo (1), o comentário apresenta uma opinião negativa sobre o console *Sony Ps3* onde o autor critica o controle do video game da *Sony*, e em seguida, elogia o controle da fabricante concorrente. Neste exemplo, existe na sentença (1) uma opinião negativa sobre o atributo *controle* do console *Sony Ps3*, e na sentença (4), outro atributo *(Wiimote)* está associado a uma palavra positiva. Por fim, é possível determinar o sentimento geral sobre cada atributo do objeto sob análise com base nas polaridades das opiniões já classificadas. Um exemplo de AS nesse nível é o trabalho [Silva et al. 2012].

2.2. Etapas da Mineração de Opiniões

Um processo completo de AS engloba quatro etapas principais: (i) detecção de subjetividade que classifica os textos entre subjetivos e objetivos, (ii) extração de atributos responsável por identificar os *atributos* do objeto em análise, (iii) classificação de sentimento/polaridade que busca determinar a polaridade do texto e (iv) apresentação dos resultados.

2.2.1. Detecção de Subjetividade

A etapa de detecção de subjetividade é responsável por identificar as sentenças subjetivas no texto. Essa etapa é fundamental para o processo de AS, pois as sentenças subjetivas apresentam de forma explícita os sentimentos: crenças e emoções sobre um determinado objeto.

Existem duas abordagens principais para a detecção automática de subjetividade na literatura: os métodos baseados em técnicas linguísticas e estatísticas ([Castro 2011],

[2]http://www.convinceme.net/debates/20/Sony-PS3-vs-Nintendo-Wii.html

[Hatzivassiloglou and Wiebe 2000]) e os métodos baseados em algoritmos de aprendizagem de máquina [Yu and Hatzivassiloglou 2003].

Como exemplo de métodos linguisticos e estatísticos, destaca-se o trabalho de [Hatzivassiloglou and Wiebe 2000]. Os autores defendem que os adjetivos são fortes indicadores de subjetividade. Dessa forma, eles determinam a subjetividade das sentenças com base na orientação semântica dos adjetivos. Em outra pesquisa, Castro (2011) realiza a detecção de subjetividade em cima de um *corpus* de *review* de filmes [Castro 2011]. Com o uso de uma ferramenta de *POS-Tagging* e o léxico *SentiWordNet* [Esuli and Sebastiani 2006], Castro (2011) identifica a subjetividade em sentenças do *corpus* e as classifica como sentenças objetivas e sentenças subjetivas alcançando taxas de acerto de até 74,5%.

Do outro lado, Yu e Hatzivassiloglou (2003) utilizaram um classificador NaiveBayes em um *corpus* composto por artigos do Wall Street Journal [Yu and Hatzivassiloglou 2003]. Através de uma combinação de atributos (e.g., unigramas, classes gramaticais e polaridades de palavras) o classificador separa sentenças objetivas (notícias e negócios) de sentenças subjetivas (artigos do editor e resposta às cartas de leitores) com precisão de 91%.

2.2.2. Extração de Atributos

A etapa de extração de atributos é responsável por identificar e extrair dos textos disponíveis os aspectos, componentes ou características associados ao objeto sob análise. Na ausência de um processo automático de extração, os atributos devem ser infomados manualmente.

Essa etapa é obrigatória quando se deseja realizar uma AS no nível de atributo, visto que na AS no nível de sentença e de documento, as características dos objetos não são analisadas individualmente.

As principais pesquisas sobre extração de atributos foram conduzidas sobre um *corpus* de *reviews* de produtos [Liu 2012]. Liu (2012) destaca que existem dois tipos mais comuns de *reviews*:

- **Tipo 1 – Pros, contras e *review* detalhado:** É pedido ao autor do comentário que descreva os atributos favoráveis e os não favoráveis separadamente, além de descrever um *review* detalhado posteriormente.
- **Tipo 2 – *Reviews* de formato livre:** O autor do comentário pode escrever livremente, sem ter que explicitar os prós e contras do produto ou serviço.

A extração de atributos nos comentários do Tipo 1 pode ser conduzida por diversos métodos (e.g., Aprendizagem de Máquina, Processamento Linguagem Natural). Este tipo de *review* geralmente consiste em frases curtas, e cada uma das sentença trata somente de um único atributo.

Os *reviews* do Tipo 2 são mais complexos para processar, pois apresentam frases completas, podendo ser encontrados vários atributos em uma única sentença. Silva (2013) foi capaz de identificar os aspectos de produtos e suas opiniões a partir de padrões linguísticos, e posteriormente classificar esses pares como positivos, negativos ou neutros [Silva 2013]. Hu e Liu (2004) descrevem uma abordagem não supervisionada para

extração de aspectos em *reviews* do Tipo 2 [Hu and Liu 2004]. Esta abordagem requer uma grande quantidade de *reviews*, e consiste em dois passos:

- **Identificar os substantivos e sintagmas nominais mais frequentes:** Hu e Liu (2004) defendem que os atributos dos objetos nos *reviews* são substantivos ou sintagmas nominais. Desta forma, o uso de um *POS-Tagger* é de grande utilidade [Hu and Liu 2004]. A razão para a extração dos substantivos mais frequentes é que, quando as pessoas comentam sobre produtos e seus atributos, o vocabulário utilizado tende a coincidir.
- **Identificar os atributos menos frequentes utilizando palavras opinativas:** As palavras opinativas geralmente são adjetivos ou advérbios que expressam opiniões positivas e/ou negativas. A justificativa para este passo é que uma mesma palavra opinativa pode ser utilizada para descrever diferentes atributos. Dessa forma, as palavra opinativas que avaliam os atributos frequentes podem avaliar os atributos não tão frequentes, portanto, podem ser utilizadas para a extração destes atributos.

Outros trabalhos que fazem extração de atributos sobre *reviews* do tipo 2 são: [Siqueira 2010] e [Lima 2011].

2.2.3. Classificação de Sentimento

Esta é a etapa do processo de AS onde são classificadas as opiniões no texto. Cada opinião possui um valor associado, que corresponde a sua orientação/polaridade (positiva, negativa ou neutra).

Aqui, também é possível se lidar com diferentes abordagens para realizar esta etapa da AS, sendo as principais delas a Aprendizagem de Máquina e a Abordagem baseada em Conhecimento, que utilizam diversas ferramentas linguísticas (e.g., *POS-Tagger*, dicionário de palavras opinativas, etc.).

Como exemplos de trabalhos que utilizam Aprendizagem de Máquina para classificação de sentimento, cita-se [Pang et al. 2002], [Wilson et al. 2005], [Mullen and Malouf 2006] e [Walker et al. 2012]. Apesar de obterem uma ótima precisão na classificação, tais soluções são muito dependentes do *corpus* de treinamento do classificador, apresentando dificuldades no processamento de textos de outros domínios.

Como alternativa, existem as abordagens baseadas em Conhecimento. Tais abordagens utilizam ferramentas linguísticas sendo os dicionários de palavras opinativas uma das principais. Estes dicionários apresentam uma lista de palavras opinativas com suas polaridades associadas. Como exemplos, citamos o SentiWordNet ([Baccianella et al. 2010], [Esuli and Sebastiani 2006]) e o léxico subjetivo MPQA [Wilson et al. 2005]. Estes métodos utilizam as polaridades associadas às palavras opinativas para determinar a polaridade das opiniões no texto.

Contudo, o uso de dicionários de palavras opinativas para a classificação de sentimento pode não ser suficiente. É importante também considerar outros elementos presentes no texto que alteram a polaridade das opiniões, como por exemplo, expressões negativas, cláusulas adversativas, concessões etc.

Exemplo 2: Comentário retirado e traduzido do debate Windows vs. Mac[3].
*"***(1)** *Eu não vou entrar em detalhes do porquê dos Macs serem melhores, porém irei dizer isso:* **(2)** *Qualquer pessoa que mudar para o Mac, Eu garanto que nunca irá voltar voluntariamente."*

Liu (2010) descreve uma abordagem genérica para esta etapa de classificação de sentimentos [Liu 2010]:

- **Identificação de palavras opinativas:** Este passo corresponde a identificação de todas as palavras opinativas presentes no texto. A seguir, para as palavras positivas, negativas e neutras são atribuídos os valores [+1], [-1], e [0] respectivamente.
- **Expressões negativas:** As expressões negativas, quando presentes no texto, invertem a polaridade das opiniões associadas a elas (e.g., *não, nenhum, nada, nunca, etc.*). Geralmente, utilizam-se valores empíricos para determinar o tamanho da janela que será utilizada para verificar se existe uma expressão negativa próxima a uma opinião. Um exemplo de sentença onde a negação não está próxima da palavra opinativa é a sentença (1) do exemplo 2.
- **Cláusulas adversativas:** Sentenças que possuem cláusulas adversativas, em geral, trazem opiniões contrárias. Dessa forma, considera-se que a polaridade da opinião antes da cláusula adversativa é contrária à polaridade da opinião depois desta cláusula. É possível averiguar uma cláusula adversativa na segunda sentença do exemplo 3.

Exemplo 3: Comentário retirado e tradozido do debate Windows vs. Mac .
*"***(1)** *Apples são bons computadores, com uma excepicional interface.* **(2)** *Vista tem melhorado sua interface, mas a Apple continua tendo a interface mais bonita, mais agradável e provavelmente será assim por vários anos."*

2.2.4. Apresentação de Resultados

A apresentação dos resultados, ou também conhecida como visualização/sumarização dos resultados, é a última etapa do processo de AS. Esta etapa é responsável por analisar os dados de saída das etapas anteriores e apresentar, de forma simples e clara, os resultados para o usuário final. As informações obtidas a partir da classificação dos comentários podem ser mostradas de diversas formas, desde textos descrevendo os resultados das análises, a gráficos que fazem comparações entre dois ou mais produtos do mesmo domínio (ver figura 1).

A figura 1 mostra um gráfico resultante da AS no nível de atributo em textos sobre duas câmeras digitais de marcas distintas. São apresentadas duas barras de cores diferentes para cada atributo das câmeras (foto, bateria, lentes, peso e tamanho), e o sentimento geral sobre os produtos - quanto mais acima da linha horizontal a barra estiver, maior é a avaliação positiva; quanto mais abaixo da linha horizontal, maior é a avaliação negativa.

Pang e Lee (2008) dividem a Visualização e Sumarização em dois tipos distintos [Pang and Lee 2008]:

[3]http://www.convinceme.net/debates/335/Windows-v-Mac.html

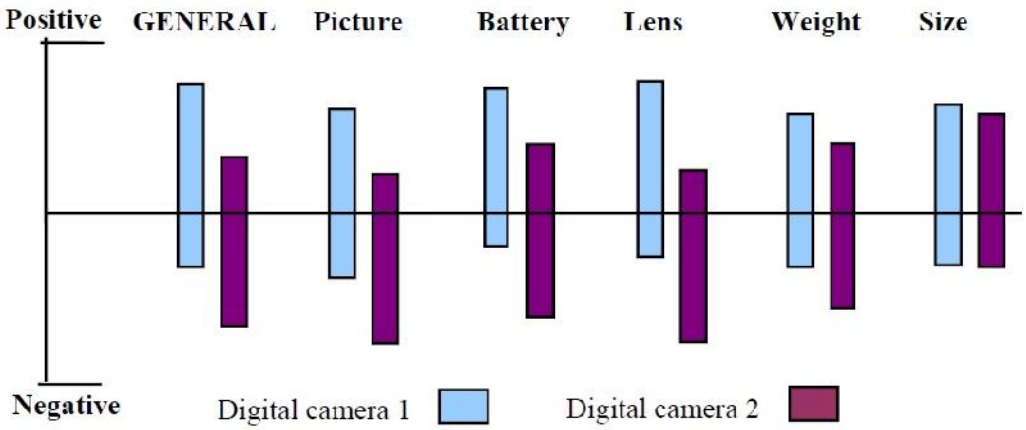


Figure 1. Resultado comparativo entre duas câmeras digitais representadas por cores distintas [Liu 2012]

- **Sumarização de documento único:** Este primeiro tipo de sumarização trata apenas das entidades de um único documento. O objetivo é apresentar o resultado da análise de sentenças extraídas, ou de unidades similares de textos, que relatam opiniões semelhantes ou que falem de um mesmo atributo, e a partir daí, apresentar os pontos positvos e os pontos negativos deste atributo.
- **Sumarização Multi-documento:** Este tipo de sumarização apresenta o resultado da análise dos objetos sobre diversos documentos de um mesmo domínio. O intuito dessa análise não é apenas mostrar ao usuário os pontos positivos e negativos de cada aspecto, como também comparar os aspectos dos dois objetos.

3. Técnicas de Mineração de Opiniões

Como pode ser visto na seção 2, a etapa de Classificação de Sentimento é responsável pela definição da polaridade no texto que é a principal tarefa da Mineração de Opiniões. Existem diversas técnicas para a classificação de opiniões, desde abordagens baseadas em algoritmos de aprendizagem de máquina [Sebastiani 2002], [Pang et al. 2002], [Walker et al. 2012], até abordagens baseadas em Conhecimento, envolvendo estatísticas e linguística [Turney and Littman 2003], [Lopes et al. 2008].

3.1. Técnicas baseadas em Aprendizagem de Máquina

A área de Aprendizagem de Máquina é uma subárea da Inteligência Artificial que estuda algoritmos e técnicas que permite o computador aprender padrões de um determinado conjunto de dados [Mitchell 1997]. Essas técnicas são dividas em aprendizado supervisionado e aprendizado não supervisionado.

As abordagens baseadas em aprendizagem de máquina apresentam bons resultados para a tarefa de classificação de sentimento ([Pang et al. 2002], [Mullen and Malouf 2006], [Thomas et al. 2006], [Abu-Jbara et al. 2012], [Somasundaran and Wiebe 2010], [Walker et al. 2012]). Contudo, na aprendizagem de máquina do tipo supervisionada, é necessário uma grande quantidade de dados etiquetados para a fase de treinamento. Esta fase é responsável pela criação de uma função de classificação que irá classificar outros dados. Dentro da área de Mineração de Opiniões,

esses dados de entrada são os textos opinativos sobre determinado domínio previamente classificados como positivo, negativo ou neutro. A fase seguinte ao treinamento dos dados (classificação), fica encarregada de receber novos textos opinativos e atribuir uma classe a esses textos, positiva, negativa ou neutra.

A utilização de algoritmos de aprendizagem supervisionada é predominante na classificação de textos [Sebastiani 2002]. No entato, estes algortimos requerem uma enorme quantidade de dados rotulados para cada domínio em que seram apliacados, o que pode ser oneroso pela escassez de dados adequados.

Pode-se citar o trabalho de [Pang et al. 2002] entre os principais trabalhos que usam essa abordagem para Mineração de Opiniões. Os autores testaram diversos algoritmos de aprendizagem supervisionada em um *corpus* de críticas de cinema, e o algoritmo que apresentou maior eficiência foi *Support Vector Machine* (SVM). Nesta Análise de Sentimento no nível do documento, Pang et al. (2002) conseguiram acurácia de 82,9%. Já Thomas et al. (2006) utilizaram o algoritmo SVM para classificar a postura de debatedores (i.e., uma opinião geral sobre uma idéia, objeto ou posição) em um *corpus* de debate do congresso americano, alcançando através dessa abordagem uma acurácia de 76,06% [Thomas et al. 2006]. Walker et al. (2012) investigaram a classificação de opiniões para descobrir a postura dos comentários de debates ideológicos (i.e., debates que possuem conotação política) e não ideológicos (i.e., debates que não possuem conotação política) [Walker et al. 2012]. Os autores utilizaram diversos algoritmos supervisionados, porém eles apenas apresentam os resultados do algoritmo *Naive Bayes*, onde seu melhor resultado foi 75,38% de acurácia.

Já nas abordagens baseadas em aprendizagem de máquina não supervisionadas não existe essa necessicade de rotulação prévia dos textos opinativos. Os algoritmos não supervisionados buscam por padrões presentes nos textos opinativos (e.g. Orientação Semântica de termos no texto), agrupando-os em classes distintas. Apesar de existirem excelentes contribuições baseadas em técnicas não supervisionadas como [Turney and Littman 2003], [Malouf and Mullen 2008], e [Abu-Jbara et al. 2012], este tipo de aprendizado não é tão utilizado como o aprendizado supervisionado em Análise de Sentimento, visto que as classes resultante dos textos em análise podem não ser previamente determinadas em uma análise manual.

3.2. Técnicas Baseadas em Conhecimento

Apesar da predominância de técnicas baseadas em aprendizagem de máquina para a classificação de sentimentos, destaca-se que, nos útimos anos, diversos trabalhos estão adotando abordagens baseadas em recursos e técnicas linguísticas para a classificação das opiniões (e.g., [Fahrni and Klenner 2008], [Somasundaran and Wiebe 2009], [Silva et al. 2012], [Ricarte Neto and Barros 2014]). Estas abordagens ainda são pouco utilizadas pois requerem uma classificação prévia das palavras opinativas para determinar a polaridade dos textos. A criação de uma base de dados com essas palavras classificadas (positivas, negativas e neutras) geralmente é feita por humanos, tornando o processo muitas vezes impraticável devido à grande quantidade de palavras opinativas que podem ser encontradas em um *corpus* de texto ou em um determinado domínio [Fernandes 2010].

Ainda assim, esses sistemas podem utilizar ferramentas linguísticas como o Sen-

tiWordNet [Esuli and Sebastiani 2006], e o dicionário de palavras opinativas MPQA [Wilson et al. 2005], e dessa forma, elimina-se a necessidade da classificação prévia das palavras opinativas. Contudo, esses métodos ainda podem encontrar dificuldades na classificação de sentimento, uma vez que algumas palavras opinativas podem ter polaridades diferentes em determinados contextos distintos.

Além dos dicionários de palavras opinativas, existem diversas outras ferramentas que auxiliam a classificação de sentimento como: *POS-Taggers*, que são ferramentas capazes de etiquetar as palavras de um dado texto com suas respectivas classes gramaticais [Toutanova et al. 2003]; Reconhecedores de Entidades Nomeadas, que são capazes de identificar quais entidades estão sendo referenciadas no texto por meio de outros termos [Finkel et al. 2005]; e até mesmo Analisadores Sintáticos, responsáveis por analisar a estrutura gramatical de uma dada frase, ou seja, quais elementos são sujeitos e objetos de um verbo, que termo é o núcleo, etc. [de Marneffe et al. 2006].

Somasundaran e Weibe (2009) usaram um analisador sintático juntamente com dicionário de palavras opinativas para auxiliar a descoberta da postura de comentários em debates de produtos [Somasundaran and Wiebe 2009]. As autoras deste trabalho relataram resultados 75% de acurácia. Já Silva et al. (2012) utiliza um *POS-Tagger* e o dicionário SentiWordNet [Esuli and Sebastiani 2006] para a classificação de sentimento à nível de atributos [Silva et al. 2012]. Os autores apresentaram resultados de 90% de precisão na classificação de sentimento. Ricarte Neto e Barros (2014) desenvolveram padrões linguísticos através de uma análise textual de textos de debates não polarizados, e que juntamente com o uso de um POS-Tagger e o dicionário MPQA [Wilson et al. 2005], buscaram determinar a postura de comentários em debates não ideológicos [Ricarte Neto and Barros 2014]. Através desse método Ricarte Neto e Barros (2014) alcançaram taxas de até 69,23% de acurácia.

4. Desafios

O crescente interesse do mercado sobre Mineração de Opiniões é devido ao potencial de suas aplicações (e.g., análise de opiniões em *reviews*, em redes sociais, em portais voltados para mercado financeiro, e em debates políticos). Igualmente importantes são os desafios que esta área apresenta para a comunidade científica. As principais dificuldades em minerar opiniões estão relacionadas aos elementos associados aos textos (e.g., erros gramaticais, erros de grafia, sarcasmos, etc).

A seguir, lista-se alguns fatores que tornam a AS um processo não trivial:

- ***Named-Entity Recognition* (NER):** Reconhecimento de entidades nomeadas é a tarefa responsável por identificar as entidades que estão sendo avaliadas no texto [Nadeau and Sekine 2007]. Em textos opinativos como *reviews* de filmes, ou de debates ideológicos/não ideológios, a NER se torna uma tarefa bastante difícil, pois as entidades aqui referenciadas podem ser de diversos tipos (e.g., pessoas, lugares, organizações). Outro fator que aumenta a dificuldade dessa tarefa são as anáforas. Nos textos opinativos, comumente os autores utilizam pronomes para fazer referência a objetos já citados. Para solucionar este problema, é necessário desenvolver um método para Resolução de Anáfora [Jakob and Gurevych 2010].
- **Dicionário de palavras opinativas:** Uma das principais fontes de erro em abordagens que utilizam dicionários de palavras opinativas são os falsos *hits* com as

palavras do dicionário [Somasundaran and Wiebe 2009]. Muitas destas palavras, de acordo com o contexto em que se encontram no texto, podem ter conotação subjetiva ou objetiva, e portanto, adicionam ruído à classificação de sentimento.

- **Detecção de Subjetividade:** Um terceiro desafio é determinar quais documentos ou partes de documentos possuem conteúdo opinativo. A detecção de subjetividade em textos extraídos de *reviews* de produtos é simples, visto que esses textos são estritamente opinativos. Já em textos extraídos de debates ideológicos, fóruns, redes sociais, a detecção de subjetividade se torna bastante complicada, pois além da estrutura do texto ser diferente dos *reviews*, é comum a ocorrência de ironias, sarcasmos, metáforas. Esses elementos podem ser erroneamente interpretados como opiniões positivas, sendo que na realidade são opiniões negativas.
- **Textos Opinativos com Ruídos:** Textos com erros de grafia, erros gramaticais, falta de pontuação e gírias ainda representam grandes desafios para os sistemas de AS. Alguns desses problemas são solucionados a partir de um bom pré-processamento no texto.

5. Conclusões

Este artigo apresenta uma visão geral de uma das áreas mais estudadas na atualidade, que é a Mineração de Opiniões. Inicialmente foi destacado algumas motivações quanto ao estudo de AS, e alguns dos conceitos principais da área, como os níveis de granularidade de sistema de AS, e as etapas de um processo completo de Mineração de Opiniões. Em seguida foi ilustrado os dois tipos de abordagens mais implementados em sistemas de Mineração de Opiniões: técnicas baseadas em algoritmos de aprendizagem de máquina; técnicas baseadas em Conhecimento que envolvem estatística e linguística. Além de mostrar conceitos relacionados a essas técnicas, foram apresentados alguns trabalhos relacionados da área de AS que utilizaram estas abordagens. Por fim, cita-se alguns desafios que fazem com que a Mineração de Opiniões seja uma das áreas mais investigadas no momento.

References

Abu-Jbara, A., Diab, M., Dasigi, P., and Radev, D. (2012). Subgroup detection in ideological discussions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 399–409, Stroudsburg, PA, USA. Association for Computational Linguistics.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Castro, A. (2011). *Detecção Automática de Subjetividade: Aprendizagem de Máquina Versus Ferramentas Linguísticas*. Trabalho de Graduação em Ciência da Computação - Centro de Informática/UFPE, Recife.

Dave, K., Lawrence, S., and Pennock, D. M. (2003). Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th*

international conference on World Wide Web, WWW '03, pages 519–528, New York, NY, USA. ACM.

de Marneffe, M.-C., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *IN PROC. INT'L CONF. ON LANGUAGE RESOURCES AND EVALUATION (LREC*, pages 449–454.

Esuli, A. and Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06*, pages 417–422.

Fahrni, A. and Klenner, M. (2008). Old wine or warm beer: Target-specific sentiment analysis of adjectives. In *Symposium on Affective Language in Human and Machine*, AISB 2008 Convention, pages 60–63, Aberdeen, Scotland.

Fernandes, F. (2010). *Um Framework para Análise de Sentimen- to em Comentários sobre Produtos em Redes Sociais*. Dissertação de Mestrado - Centro de Informática/UFPE, Recife.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hatzivassiloglou, V. and Wiebe, J. M. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, COLING '00, pages 299–305, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Jakob, N. and Gurevych, I. (2010). Using anaphora resolution to improve opinion target identification in movie reviews. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 263–268, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Jindal, N. and Liu, B. (2007). Review spam detection. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1189–1190, New York, NY, USA. ACM.

Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lima, D. (2011). *PairExtractor: Extração de Pares Livre de Domínio para Análise de Sentimentos*. Trabalho de Graduação em Ciência da Computação - Centro de Informática/UFPE, Recife.

Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, 2nd ed.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 1 edition.

Lopes, T. J. P., Hiratani, G. K. L., Barth, F. J., Rodrigues, Jr., O., and Pinto, J. M. (2008). Mineração de opiniões aplicada à análise de investimentos. In *Companion Proceedings of the XIV Brazilian Symposium on Multimedia and the Web*, WebMedia '08, pages 117–120, New York, NY, USA. ACM.

Malouf, R. and Mullen, T. (2008). Taking sides: User classification for informal online political discourse. Internet Research.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition.

Mullen, T. and Malouf, R. (2006). A preliminary investigation into sentiment analysis of informal political discourse. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW*, pages 159–162.

Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proc. of the ACL-02 Conf. on Empirical Methods in Natural Lang. Proc. - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Ricarte Neto, F. A. and Barros, F. (2014). Asdp: um processo para análise de sentimento em debates polarizados. In *Encontro Nacional de Inteligência Artificial e Computacional*, BRACIS/ENIAC 2014, pages 1–6, São Carlos, SP, Brasil. Proc. of the Brazilian Conference on Iteligent Systems.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47.

Silva, N, G. (2013). *PairClassif - Um Método para Classificação de Sentimentos Baseado em Pares*. Dissertação de Mestrado - Centro de Informática/UFPE, Recife.

Silva, N., Lima, D., and Barros, F. (2012). Sapair: Um processo de análise de sentimento no nível de característica. In *IV International Workshop on Web and Text Intelligence (WTI – 2012)*, pages 1–10, Curitiba, PR, Brasil. Proc of the Brazilian Conference on Iteligent Systems.

Siqueira, H. (2010). *WhatMatter: Extração e visualização de características em opiniões sobre serviços*. Dissertação de Mestrado - Centro de Informática/UFPE, Recife.

Somasundaran, S. and Wiebe, J. (2009). Recognizing stances in online debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1*, ACL '09, pages 226–234, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Somasundaran, S. and Wiebe, J. (2010). Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 116–124, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Thomas, M., Pang, B., and Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 327–335, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proc. of the 2003 Conf. of the North American Chapter of the Assoc. for Comput. Ling. on Human Lang. Tech. - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346.

Walker, M. A., Anand, P., Abbott, R., Tree, J. E. F., Martell, C., and King, J. (2012). That is your evidence?: Classifying stance in online political debate. *Decis. Support Syst.*, 53(4):719–729.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA. Assoc. for Comput. Linguist.

Yu, H. and Hatzivassiloglou, V. (2003). Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 129–136, Stroudsburg, PA, USA. Association for Computational Linguistics.

Promovendo sua ideia para um modelo de negócio

Caio Farias Bitterncourt[1]
Patrick Mazulo de Brito[2]
Welk de Oliveira Silva[3]

Resumo: Em virtude que este seguimento empreendedor tem ganhado cada vez mais espaço dentro do contexto tecnológico, o trabalho proposto tem como objetivo abordar de forma simples os passos para a construção de uma solução viável e a validação de conversão em um modelo de negócio escalável a partir de uma simples ideia, seguindo como base a metodologia LEAN STARTUP.

Palavras-chave: Empreendedorismo, Lean Startup, Startup.

Abstract: *Due to this entrepreneurial segment has gained increasing space within the technological context, the proposed work aims to address the simple steps to building a viable solution validation and conversion into a scalable business model from a simple idea, based on the following methodology LEAN STARTUP..*

Keywords: *Entrepreneurship*, Lean Startup, Startup.

1 Introdução

O termo Empreendedorismo, apesar de ser popularizado, não existe um conceito padrão, mas há definições citadas por especialistas da área, como Joseph Schumpeter (1950) que aponta a uma pessoa com criatividade e capaz de fazer sucesso com inovações, assim como K. Knight (1967) e Peter Drucker (1970), que de uma forma geral introduz o conceito de risco para a pessoa empreendedora. Baseado no que grandes nomes do empreendedorismo definem, podemos concluir que empreendedorismo é o movimento que introduz novos produtos e serviços, criando novas formas de organização ou explorando novos recursos e matérias.

Partindo agora para a visão do mercado empreendedor, podemos observar que o mesmo já atinge os mais diversos setores da economia. Porém na Tecnologia da Informação não poderia ser diferente, pois neste setor é que se observa um dos maiores crescimento de inovação empreendedora. Podemos citar empresas que comumente partiram de um mesmo princípio inovador e investindo em riscos como Dropbox, Easy Taxi, Boo Box, Urbano, Net Flix, e não menos notável o grande Facebook, que partiu de uma ideia bastante empreendedora em que se torna uma das mais ricas empresas da internet.

Associado ao conceito de empreendedorismo, surge em meio às pequenas e potenciais empresas com novas ideias o termo Startup, que identifica propostas tecnológicas inovadoras, tímidas, de caráter experimental, de pequeno porte e de pequeno investimento financeiro, mas com um grande valor intelectual investido.

O início de uma Startup aliado ao comportamento empreendedor nasce principalmente a partir de uma ideia, mas para se ter uma ideia, deve haver algo motivador, que seria o problema. Esse conceito de que toda ideia parte de um problema, é bem comum, mas para Steven Johnson (2010), um dos grandes pensadores da

[1] Universidade Estadual do Piauí (UESPI) Caixa Postal 64200-000 – Parnaíba – PI – Brasil
{caicarybio@gmail.com}
[2] Faculdade Maurício de Nassau – Campus Parnaíba - Caixa Postal 64200-000 – Parnaíba – PI – Brasil
{pmazulo@gmail.com}
[3]Universidade Estadual do Piauí (UESPI) Caixa Postal 64200-000 – Parnaíba – PI – Brasil
{welksilva@gmail.com}

internet, aborda em seu livro “De onde vêm as boas ideias” um conjunto de observações apontam o nascimento de uma boa ideia, tais como: “Grandes ideias não surge de repente, elas levam um tempo para amadurecer”, “Grandes ideias pode ser a junção da várias pequenas ideias”, “O acaso favorece a mente conectada”.

Portanto, ter uma boa ideia nada mais é do que estar iterado ao meio que se busque uma solução. A partir dos próximos tópicos será apresentado a forma de como valorizar, validar, iterar e formalizar sua ideia além de minerar ideais complementares.

2 Conceitos básicos sobre o Lean Canvas

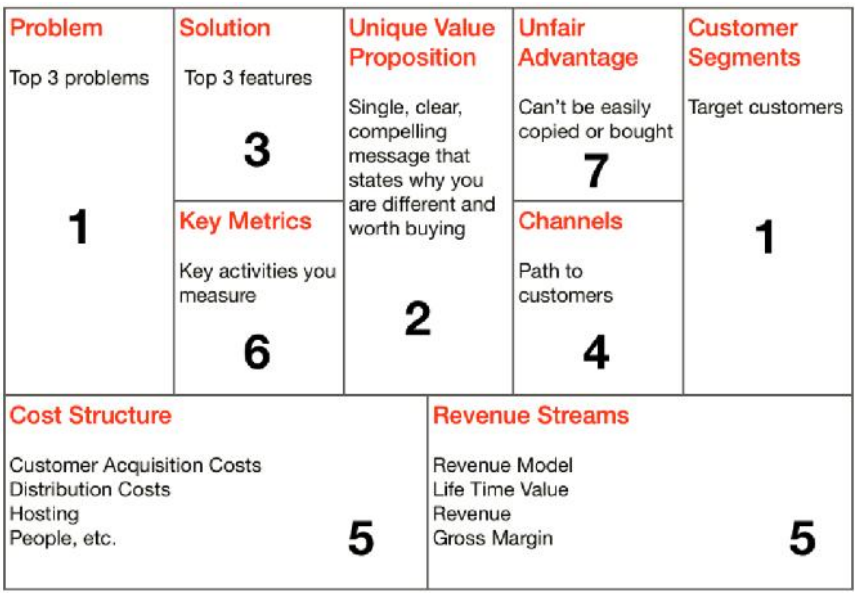
Em meio a essa grande quantidade de ideais e informações, observa-se a necessidade da aplicação de uma maneira para otimizar o processo de amadurecimento de uma ideia. O *Lean Canvas* é uma adaptação do BMC (*Business Model Canvas*) de Alexander Osterwalder, e criada por AshMaurya, que aborda exatamente este ponto, sendo esta uma técnica que auxilia na definição e no refinamento, de forma ágil, um plano de negócios, sendo ele a criação de uma nova empresa ou de um produto inovador. Sendo assim proposto para auxiliar na evolução rápida das ideais iniciais de um novo projeto, com o objetivo de potencializar o aprendizado prático.

Assim como o BMC, o *Lean Canvas* também é dividido em nove blocos (Problema, Segmentos do Cliente, Proposição Única de Valor, Solução, Vantagens Injustas, Fontes de Receitas, Custo da Estrutura, Métricas Chaves e Canais), porém a estratégia de preenchimento é diferente. É possível entender melhor observando a Figura 1.

2.1 Problemas e Segmentos dos Clientes

Serão completados inicialmente os blocos “Problemas” e “Segmentos de Clientes”, pois segundo Maurya (2010, p.28) eles darão o embasamento necessário para que se possa prosseguir, ou não, dos outros blocos. No bloco “Problemas, são listados os principais (de preferências 3) causados aos clientes. A seguir descobre-se quais as maneiras que o cliente usa para resolver tais problemas, sabendo assim qual o nível da “dor” do cliente, e se seria o suficiente para este aceitar uma nova solução.

Figura 1: Quadro do *Lean Canvas*



2.2 Proposição Única de Valor

Finalizando este preenchimento inicial, passa-se a pensar sobre a definição da PUV, Proposição Única de Valor (ou Proposta de Valor), sendo esta uma das fases mais importantes e difíceis deste *Canvas*, diz Maurya (2010, p.50). A PUV é difícil de ser definida porque nesta terá que conter em poucas palavras a essência do seu produto, ou serviço, para que possam também caber dentro da chamada de uma futura *Landing Page*. A PUV tem que focar nos "adotantes primários" (em inglês, early adopters), mostrando os reais benefícios do objetivo pretendido (produto ou serviço).

2.2 Solução

Ao ter em mente o problema e o nível de dor que este causa ao seu cliente, o segmento de clientes e proposta de valor, o próximo é pensar sobre as soluções que o seu produto ou serviço irá oferecer para os seus clientes. Da mesma maneira como em "Problemas", você irá esboçar de maneira simples as três características ou habilidades que solucionam cada um dos 3 problemas percebidos anteriormente.

2.4 Canais

AshMaurya também defini no seu quadro canais de comunicação entre a Startup e os clientes, ou seja, o caminho que você deverá percorrer para chegar até seus clientes. Canais são normalmente associados a coisas como SEO, Redes Sociais e Blogs, ou seja, tem um custo zero de capital humano. Há uma série de variáveis que devem ser muito bem calculadas nessa parte, como por exemplo: Qual público alvo? Quanto de recurso tenho disponível? Grátis vs Pago? Como será este Marketing de Conteúdo?

2.5 Estruturas de Custos e Fluxos de Receita

"Agora chega a hora de preencher o que se torna a base para a viabilidade do projeto, sendo eles as "Estruturas de Custos" e Fluxos de Receita". Nas Estruturas de Custos serão avaliados desde gastos com recursos humanos, até tecnológicos. É quanto será gasto para que seu produto ou serviço fique disponível para seus clientes. O Fluxo de Receita tem influência em uma série de fatores, e entre eles o mais importante é o Produto Mínimo Viável (em inglês, Minimum Viable Product - MVP), pois este deve abordar os principais problemas que os clientes tem mostrado como importante para eles, para que esteja justificada uma possível cobrança pelo produto ou serviço prestado. "Seu preço faz parte do seu produto." Maurya (2010, p.50).

2.6 Métricas Chaves

No nosso processo do *Lean Canvas*, temos este ponto importante e extremamente relevante para as Startups. São estas métricas que irão identificar o que medir para verificar a performance do projeto. Maurya aconselha o uso da técnica das Métricas Piratas de Dave McClure's ilustrada pela Figura 2, onde você tende a responder perguntas como: Como os usuários encontrarão você? Os usuários terão uma boa primeira experiência? Estes usuário irão voltar? Como você fará dinheiro? Os usuários falarão para outros?

Figura 2: Métricas Piratas

2.7 Vantagens Injustas

Por último a ser preenchida, temos as Vantagens Injustas, sendo considerada como a mais difícil de ser feita. Maurya diz que é necessário ter muita cautela, pois nada impediria que uma empresa endossadamente mais especialista entre no mesmo mercado que o escolhido para o projeto, e o faça sem cobrar nada, fidelizando assim seus possíveis clientes. Então esse bloco deve ser o diferencial, fazendo com que nenhuma outra empresa mesmo sendo maior, consiga realizar seu projeto antes de você, ou que pelo menos a atrase bastante. Uma excelente equipe, um grande feito nas redes, uma comunidade envolvida no projeto e um bom posicionamento em sites de buscar como vantagens primordiais para serem avaliadas nesta batalha.

3. Validação

“A vida é muito curta para construirmos algo que ninguém quer.” - AshMaurya. Muitas empresas e empreendedores já constataram a importância da validação de suas ideias. Sem dúvidas, o primeiro passo para o fracasso é começar a desenvolver algo apenas com suas próprias visões do negócio. Através da validação junto com seu público-alvo, conseguiremos a resposta sobre a adoção por parte dos usuários, conhecendo sua real necessidade diante do problema que o projeto pretende resolver, bem como a viabilidade do mesmo.

A primeira forma de validação que vem a mente é usar os típicos formulários estruturados e envia-los pela web (Google Docs, email, Twitter e Facebook, por exemplo). Dessa forma, você cria um formulário engessado, com sua visão do negócio, sem chance alguma do cliente expor sua opinião do seu próprio problema, indicando soluções que podem servir como base para as soluções dos problemas. Nem mesmo deixando campos abertos para o usuário expor sua opinião que é a melhor opção, pois acaba se tornando uma atividade enfadonha.

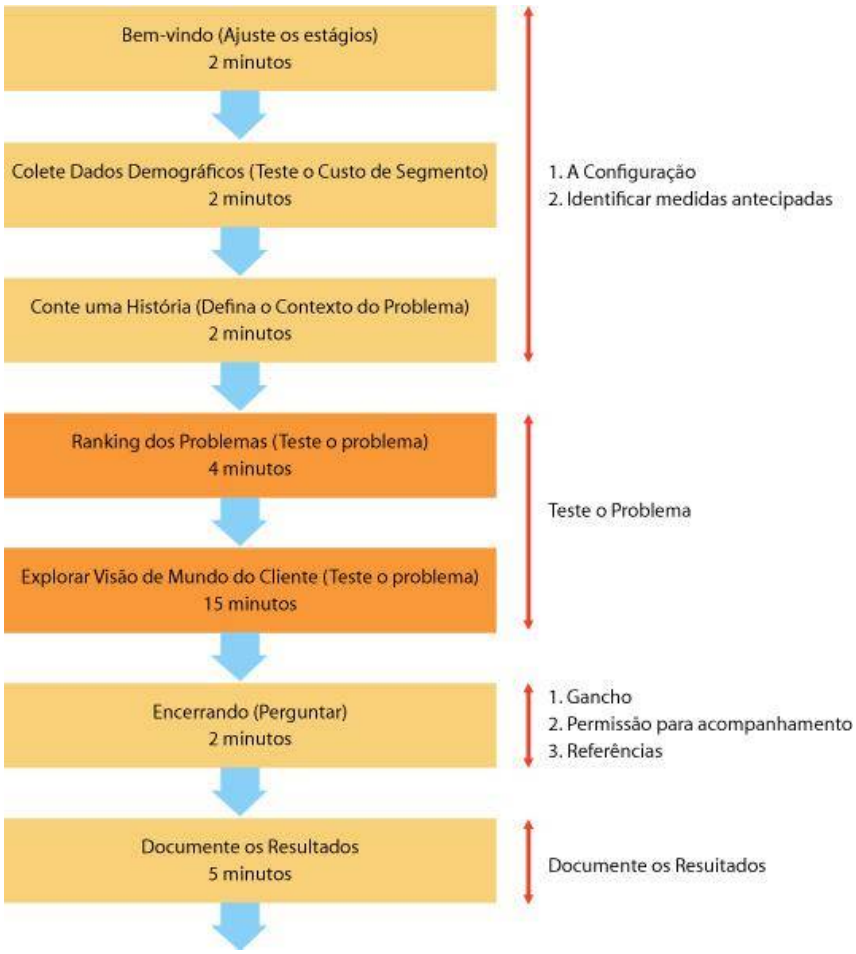
De tal maneira, fica claro que essas ferramentas devem ser evitadas. Mas que rumo seguir para concluir a validação? A maneira mais eficiente se mostra através de uma entrevista presencial. Dessa forma, além de passar

uma imagem de interesse e comprometimento para seu possível futuro usuário, terão de forma rápida e direta as informações pertinentes sobre a equação problema/solução e o nível da dor sofrida pelo entrevistado. Na Figura 3 é possível acompanhar as etapas da entrevista, bem como a organização do tempo para cada etapa.

Porém antes de partir para a entrevista, é necessária uma preparação, tendo como critérios: Montar um bom time, criar hipóteses mensuráveis, priorize os problemas de acordo com a sua visão, monte a primeira lista de potenciais entrevistados, crie seu roteiro e relatório. Ao preparar estas variáveis, o próximo passo é organizar e preparar-se para as entrevistas. As entrevistas devem ser preferencialmente presenciais, sendo que o objetivo não é saber o que o entrevistado quer, mas o que ele faz para resolver seu problema. Uma dica para o início dessas entrevistas seria começar com pessoas conhecidas, dando assim à equipe entrevistadora uma maior prática e habituação. Dê preferência a um local neutro, com um clima mais informal, no intuito de deixar o entrevistado o mais confortável possível, e evite ao máximo extrapolar o tempo combinado entre as partes. E o mais importante: documente a entrevista logo após o término, para não correr o risco de esquecer nenhuma informação.

Usando destes artifícios, o resultado será a construção de um contexto de aprendizado sobre o problema do usuário, bem como identificar os *early adopters*. Com essas informações também será possível validar os blocos "Segmentos de Clientes" e "Problemas".

Figura 3: Mapa para estruturação de uma boa entrevista



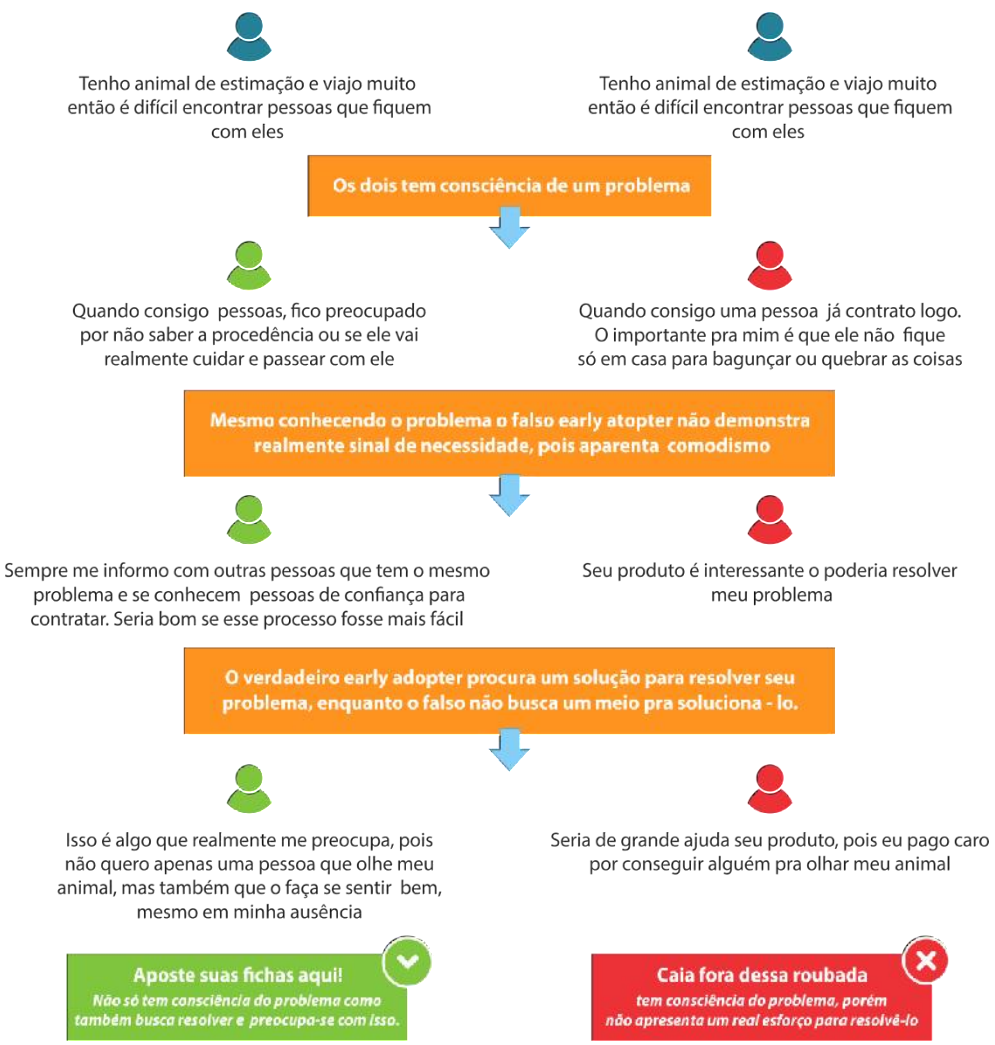
4. Garimpando os verdadeiros *early adopters*

A identificação dos seus *early adopters* (Primeiros usuários) é de decisiva importância para a escalabilidade de sua Startup, pois são eles os que mais se sentem incomodados com o problema e procuram resolver mesmo que não haja ainda uma solução realmente viável.

São essas pessoas que realmente contribuirão para suas entrevistas de refinamento do projeto, pois elas não apenas responderão sinceramente aos seus questionamentos como também abrirão possíveis campos não explorados antes na solução inicial. Mantenha total atenção ao que elas dizem de forma a capturar quais problemas elas enfrentam e o que fazem atualmente para resolvê-los. Questione qual a etapa mais complicada do processo de resolução do problema e como seria ideal para ameniza-lo ou até mesmo soluciona-lo de fato.

Os *early adopters* serão peças fundamentais para o lançamento e escalabilidade de sua Startup, todavia que eles darão inicio a disseminação do seu produto/serviço através dos testemunhos de aprovação, pois são mais valiosos que o marketing da Startup. A Figura 4 demonstra um exercício de abstração para se identificar um verdadeiro ou falso *early adopter*, observe que durante o processo de entrevista há uma diferença de medir esforços para solucionar ou não o problema.

Figura 4: exemplo de abstração de um verdadeiro *early adopter*

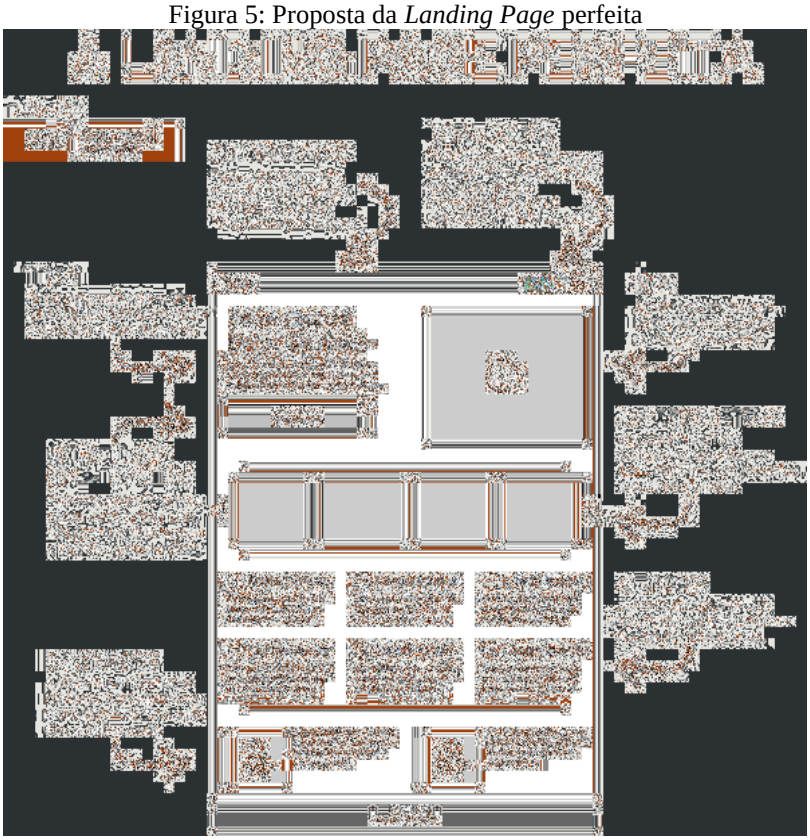


5. *Landing Pages*

A busca por um maior número de aceitação dos potenciais clientes antes do lançamento da aplicação é a construção de uma *Landing Page*. Este é um importante passo para a divulgação.

Na Figura 5 é possível ver um gráfico apresentado pelo site Idie Game Girl, que exprime a ideia da *Landing Page* perfeita, observe os elementos e as disposições dos mesmos.

Figura 5: Proposta da *Landing Page* perfeita



Fonte: http://www.indiegamegirl.com/landing-page-design-and-how-to-use-it-to-sell-your-indie-game/

É certo que este elemento é um dos primeiros a serem observados pelo usuário, portanto é preciso que o mesmo atraia e segure a atenção e curiosidade. Segundo Henrique Carvalho, criador do site Viver de Blog, destaca alguns aspectos importantes para a criação de títulos: que atraia à atenção, que desperte o interesse no assunto e o desejo de ler, e que estimule uma ação em seguida que seja útil ao leitor, que cria um senso de urgência, que transmita a ideia de unicidade, e que seja o mais específico possível.

O objetivo principal de uma *Landing Page* é proporcionar o conhecimento e aceitação do produto apresentado, porém esses critérios devem ser feito por partes definidas pelo criador da página, portanto um menu de navegação que leve o usuário à partes da página irá desorganizar a ordem prevista pelo autor.

A intenção da divulgação é a expansão da informação do produto disposto na *Landing Page*, então é importante a existência de ícones sociais, preferencialmente na parte superior da página, para o compartilhamento e claro, a

aprovação do usuário. Ainda é importante ressaltar a necessidade de um botão que leve o usuário a uma interação mais próxima ao seu produto, tais como: “Baixe agora”, “cadastre e receba”, “Experimente agora”.

Elementos visuais sempre caem bem em qualquer apresentação, portanto na página do seu produto não poderia ser diferente. É de grande valia que se tenha um vídeo explicativo que narre e anima situações do produto, até mesmo algumas das formas de usar. Imagens também valorizam a *Landing Page*, além das que inspiram a ideia, é essencial que tenham *screehshoots* da aplicação para que o usuário se familiarize como produto.

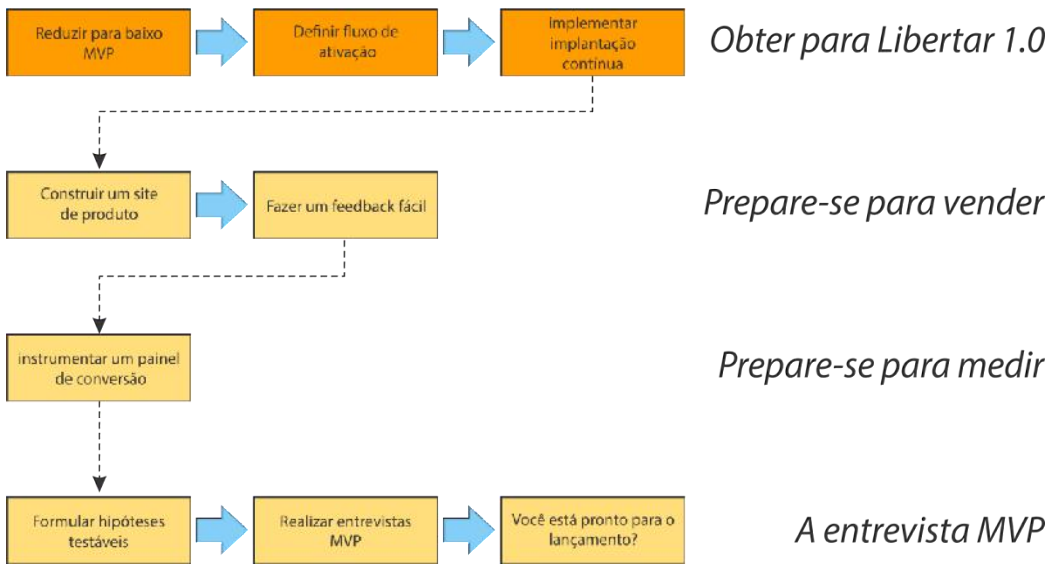
6. Minimum Viable Product - MVP

Após os passos de planejamento citados acima é dado início a execução do protótipo, denominado MVP, que será o produto minimamente viável.

A definição de MVP vai mais além de um simples protótipo, pois ele será um produto real e funcional para atender as mais importantes necessidades da solução proposta. Reduzindo o escopo que o MVP ira cobrir, será encurtado não somente o ciclo de desenvolvimento, mas também irá remover distrações desnecessárias que irão diluir a mensagem que o seu produto ou serviço querem passar.

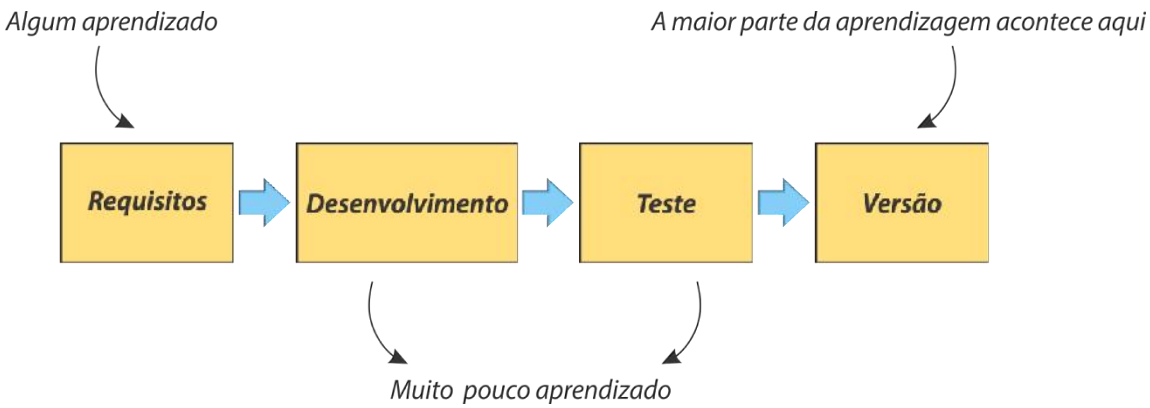
Além de priorizar os requisitos certeiros, o mesmo servirá para abstrair do usuário se a necessidade principal está sendo suprida, além de economizar tempo no processo de construção da solução para que posteriormente seja implantando os suplementos que o usuário realmente utilizará. Enquanto alguns aprendizados ocorrem durante a fase de levantamento de requisitos e entrevistas, a maior parte dele virá depois de lançado o produto.

Figura 6: Mapa para a construção e lançamento do MVP



Os passos para a construção do MVP devem ser baseados em um levantamento minucioso e o mais preciso possível à realidade da necessidade do cliente. Para facilitar este processo que pode acabar se tornando muito complexo, podemos definir 5 etapas importantes para o desenvolvimento do MVP.

Figura 7: Ciclo de desenvolvimento do produto



6.1 Limpe seu quadro

Não assuma que todas as funções deverão automaticamente ser incluídas no seu MVP. Inicie com um quadro em branco, e justifique a adição de cada função. Deve-se lembrar que MVP é o Produto Minimamente Viável.

6.2 Comece com seu problema número 1

O trabalho da sua Proposta de Valor é fazer uma promessa atraente para o problema do seu cliente. Algo que de frente ao nível de dor que este sente para normalmente solucionar seu problema, seja de extrema necessidade. O trabalho do MVP é entregar e cumprir com esta promessa. A essência do seu MVP deveria ser capturada no *mock-up* do seu problema número 1. Comece por aqui.

6.3 Elimine nice-to-haves e don't-needs

Depois do processo de entrevistas, você deverá ser capaz de nomear cada elemento no seu *mock-up* como *must-have* (preciso dessa solução), *nice-to-have* (seria bom ter essa solução) e *don't-need* (não preciso dessa solução). De imediato elimine todos os *don't-needs*, e adicione os *nice-to-haves* na sua fila de *backlog* como futuras funções, a não ser que alguma dela seja pré-requisito para uma característica *must-have*. Repita este passo para os seus *mock-ups* dos problemas número dois e três.

6.4 Considere as solicitações de recursos de clientes

Seus clientes podem ter dado destaque em algum recurso para fazer o seu produto, ou serviço, algo realmente útil e completo, como uma integração de *login* com Facebook, por exemplo. Reveja cada uma destas solicitações, e então adicione-as de acordo com o nível de necessidade (*must-have, nice-to-have e don't need*).

6.5 Foco no aprendizado, não na otimização

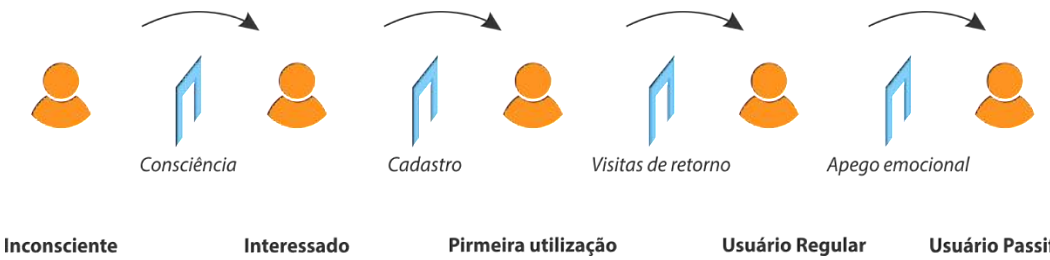
Toda energia e atenção precisam ser canalizadas para o aceleramento da aprendizagem. Velocidade é a chave mestra para isso. Não perca tempo ou gaste esforço tentando otimizar servidores, códigos e bancos de dados.

Deixe essas tarefas para o futuro. Existem grandes chances de você não ter um problema com solução escalável quando lançar o MVP. Mas não é nada que não seja resolvido com um hardware adicional, o que até mesmo justifica a cobrança para seus clientes, para que a solução do problema seja de forma mais eficiente.

7. Entendo o Ciclo de Vida do Usuário

Ao finalizar os requisitos mais importantes, ou seja, fechar o escopo do seu MVP, chega-se a hora de lançar seu produto, e validá-lo. Sabendo disso devemos ter em mente mais uma coisa: enquanto construir alguma coisa que as pessoas querem é o objetivo final da Startup, você não pode deixar de realmente entender seus usuários. Quando você lança um produto, muitas coisas podem e vão dar errado. Muitas dessas não necessariamente têm a ver com o produto. Por exemplo, você pode estar perdendo pessoas apenas pelo posicionamento, design ou mesmo preço. Este é o aprendizado que o MVP traz sob o usuário.

Figura 8: Modelo do ciclo de vida por Joshua Porter



O Ciclo de Vida do Usuário descreve o caminho que um usuário leva ao estar em uso do seu produto, e eventualmente tornar-se um usuário “apaixonado”. Existem alguns modelos que ajudam a entender este ciclo, sendo um destes: O Uso do Ciclo de Vida. Este modelo foi construído por Joshua Porter, e captura qualitativamente o ciclo de vida do usuário da perspectiva do próprio usuário. Ele se concentra em entender a motivação dos seus usuários, e as barreiras psicológicas para que sejam tomadas posições e decisões concisas para superar as mesmas.

Referências

MAURYA, Ash. Running Lean: Iterate from Plan A to Plan That Works. 1ª edição. 2010.

MAURYA, Ash. Running Lean: Iterate from Plan A to Plan That Works. 2ª edição. 2012.

Modelo Lean Canvas

http://startupbizmodel.com/2011/08/09/lean-canvas/

Lean Canvas na Prática: Resumo comentado do Workshop de Ash Maurya

http://www.infoq.com/br/articles/resumo-workshop-lean-canvas

OLIVEIRA, Fabiana Morais. Epreendedorismo: Teoria e Prática. Especialize IPOG. Pag 1. Maio/2012

http://viverdeblog.com/como-escrever-titulos/

http://www.indiegamegirl.com/landing-page-design-and-how-to-use-it-to-sell-your-indie-game/

Testes de Invasão: Metodologia, Técnicas e Ferramentas

Felipe Santos Barbos[1]

[1]Departamento Nacional de Infraestrutura e Transportes – DNIT/PI

felipe.barbosa@dnit.gov.br

Abstract. *The main objective of this paper is to simulate a controlled manner a real attack that is usually run by criminals. This way you can have the full knowledge of what could happen if the attack actually was, thus ensuring the possibility of a prevention strategy. Where are differentiated the concepts of Assessment and Pentest, presenting basic concepts of Information Security and approaches Security Test failures of systems or infrastructure.*

Resumo. *O objetivo principal deste artigo é simular de forma controlada um ataque real que normalmente é executado por criminosos. Desta maneira é possível ter o conhecimento total do que poderia acontecer caso o ataque realmente existisse, garantindo assim a possibilidade de uma estratégia de prevenção. Onde são diferenciados os conceitos de Assessment e Pentest, apresentando conceitos básicos de Segurança da Informação e abordagens de Teste de Segurança as falhas de sistemas ou infraestrutura.*

1. Introdução

A principal diferença entre Assessment e Pentest é que este último vai além de identificar vulnerabilidades, partindo para o processo de exploitation, escalada de privilégios e garantia do acesso ao sistema alvo. O Assessment apresenta uma visão mais abrangente sobre as falhas de sistema ou infraestrutura. Já o Pentest tem como objetivo direto testar as políticas e implementações de segurança em uma organização. Além disso o Pentest é muito mais intrusivo.

Os testes de invasão podem ser definidos como uma tentativa legal e autorizada de localizar e explorar sistemas de computadores de forma bem-sucedida com o intuito de tornar esses sistemas mais seguros. O processo inclui sondar as vulnerabilidades, bem como oferecer ataques que funcionem como prova de conceito para demostrar que eles são reais.

Os testes de invasão adequados sempre terminam com recomendações especificas para endereçar e corrigir os problemas descobertos durante o teste. Esse processo como um todo é usado para ajudar a manter as redes e os computadores seguros contra ataques no futuro.

A ideia geral consiste em identificar problemas de segurança usando as mesmas ferramentas e técnicas usadas por um invasor. Podendo então atenuar os riscos identificados por essas descobertas antes que um hacker de verdade os explore.

Os testes de invasão também são conhecidos como:

PT (penetration testing)

Hacking ético

Hacking White Hat

Segurança Ofensiva

2. Segurança da Informação

Segurança da Informação refere-se a proteção das informações contidas no domínio das empresas ou pessoas. Isto envolve qualquer ativo que gere, processe, manipule, transmita ou armazene informações. Tem como requisito a garantia de continuidade de negócio, minimizando os riscos e maximizando o retorno sobre os investimentos. É constituída de um conjunto de controles, incluindo políticas, normas, padrões e procedimentos. (SEMOLA, 2002).

3. Princípios básicos da Segurança da Informação

A norma NBR ISO/IEC 27001 estabelece os seguintes atributos básicos, também conhecidos como pilares da segurança da informação com o acréscimo de mais duas, que permitem a troca segura de informação, desde que nenhum deles seja violado. São eles:

3.1. Confidencialidade

Esse pilar é o responsável pelo controle de acesso à informação apenas por aquelas pessoas ou entidade que tenham permissão compatível com sua função e determinada pelo dono daquela informação.

3.2. Integridade

Aqui, através dessa propriedade, é determinada a necessidade de garantir que a informação mantenha todas as suas características originais como determinadas pelo proprietário da informação.

3.3. Disponibilidade

Propriedade que define que determinada informação esteja sempre disponível para o acesso quando necessário, de maneira íntegra e fidedigna. Alguns dos ataques conhecidos buscam justamente derrubar a disponibilidade, e para algumas empresas o simples fato de não ter suas informações disponíveis durante determinado período de tempo, isso pode acarretar prejuízos estrondosos.

3.4. Autenticidade

Propriedade responsável por garantir que a informação vem da origem informada, permitindo a comunicação segura e garantia de que a informação a qual tem acesso é correta e de fonte confiável.

3.5. Legalidade

É a propriedade que define se determinada informação, ou operação, está de acordo com as leis vigentes no país. As mesmas leis que regem um país podem ser completamente diferentes em outro, o que pode ocasionar uma série de problemas, caso o sistema de gestão não seja adaptável. (TI EX. ISO/27002 FOUNDATION, 2011).

Podemos ver na figura a seguir alguns dos distúrbios mais comuns aos pilares da SI, vinculados a ataques que visam à área de TI:

Abrangência	Ameaça	Exemplo
Confidencialidade	*Browsing*	Procurar por informações sem necessariamente saber seu tipo
	Shoulder surfing ("papagaio de pirata")	Olhar sobre o ombro da pessoa o que é digitado.
	Engenharia Social	Fingir ser alguém com a intenção de ter acesso a informações.
Integridade	Modificar uma mensagem	Interceptar uma mensagem, alterá-la e enviá-la ao seu destino original
	Alteração de logs de auditoria	Modificar os logs de auditoria, normalmente com a intenção de ocultar fatos
	Modificação de arquivos de configuração	Alterar arquivos críticos em um sistema para modificar sua fucionalidade.
Disponibilidade	Desastres naturais ou provocados	Vandalismo, incêndios, terremotos, terrorismo, vulcanismo, ...
	Negação de serviço (DoS)	Comprometimento de serviços de importância fundamental para processos
	Comprometimento de informações	Modificar dados de forma a ficarem inúteis para outras pessoas

Figura 1. Pilares da Segurança da Informação

4. Terminologias de Segurança

- **Vulnerabilidade:** fragilidade que pode fornecer uma porta de entrada para um atacante.
- **Ameaça:** agente ou ação que se aproveita de uma vulnerabilidade.
- **Risco:** (Impacto X Probabilidade) da ameaça ocorrer.
- **Ataque:** incidência da ameaça sobre a vulnerabilidade.
- **Exploit:** programa capaz de explorar uma vulnerabilidade.

5. Serviços de Segurança

Existe hoje em dia um elevado número de ferramentas e sistemas que pretendem fornecer segurança. Alguns exemplos são os detectores de intrusões, os antivírus, firewalls, firewalls locais, filtros AntiSpam, fuzzers, analisadores de código, etc.

Além de dispositivos de segurança, também existem diversos serviços relacionados à segurança da informação.

Esses serviços precisam de profissionais com um conhecimento altamente especializado, primeiro por lidar com análises complexas, e segundo por envolver informações sigilosas que precisam de tratamento especial, para que não sejam comprometidas de alguma maneira.

Dentre os serviços oferecidos por profissionais de segurança estão:

Criação de Políticas de Segurança;

Hardening de Servidores;

Análise de Vulnerabilidade;

Teste de Invasão;

Perícia Computacional;

Treinamento de Colaboradores;

Auditoria.

6. Visão geral sobre o Pentest

O Teste de Intrusão é um processo de análise detalhada do nível de segurança de um sistema ou rede usando a perspectiva de um infrator. Trata-se de um teste realista ao nível de segurança das infraestruturas e da informação que estas detêm. No Teste de Intrusão são testadas vulnerabilidades técnicas e conceituais das infraestruturas alvo.

7. Diferença entre Testes de Invasão e Avaliação de Vulnerabilidades

Pessoas e fornecedores na comunidade de segurança usam esses termos de forma incorreta. Uma avaliação de vulnerabilidade corresponde ao processo de analisar serviços e sistemas em busca de problemas de segurança em potencial, enquanto um teste de invasão realmente executa explorações de falhas (exploitation) e ataques como prova de conceito para provar a existência de um problema de segurança. Os testes de invasão vão um passo além das avaliações de vulnerabilidades, simulando a atividade de um hacker e enviando payloads ativos. (ENGEBRETSON, 2014).

8. Tipos de abordagens de Testes de Segurança

Há basicamente três abordagens, Black-Box, White-Box, Gray-Box, onde (Tandem e Reversal) são abordagens que fazem parte de acordo com a metodologia a ser utilizada no pentest. Tais abordagens são descritas nas sessões que seguem.

8.1. Black-Box

Neste caso assume-se que o auditor de segurança inicia o teste de segurança sem nenhum conhecimento do ambiente a ser testado e de sua infraestrutura. (RAMOS et al, 2008, p.279). Assim é possível descobrir conjuntos de vulnerabilidades conhecidas e também não conhecidas (0 day). O auditor que executa testes Black Box é conhecido como Black-Hat. Este tipo de teste é também conhecido como teste externo.

Após a realização dos testes e classificação do nível de severidade das vulnerabilidades é gerado um relatório, que mostre o grau de exposição de determinado ativo ou de uma empresa inteira, dependendo do escopo da avaliação.

8.2. White-Box

Neste caso assume-se que o auditor de segurança possui toda a informação sobre os sistemas-alvo, topologias de rede, diagramas com nomes e endereços IP dos hosts que serão testados, etc. (RAMOS et al, 2008, p.280). Há um esforço muito menor para descobrir vulnerabilidade neste caso. Este tipo de teste também conhecido como Teste Interno. O auditor que executa testes White-Box é conhecido como White-Hat.

O teste White-Box pode ser facilmente integrado em um ciclo de desenvolvimento ou manutenção regular para mitigar quaisquer problemas de segurança em sua fase inicial antes de serem divulgados e explorados. O tempo e o custo necessário para encontrar e solucionar vulnerabilidades de segurança é menor do que na abordagem Black-Box.

8.3. Gray-Box

Os dois tipos de testes anteriores combinados oferecem uma estratégia bastante eficiente do ponto de vista da segurança. O auditor envolvido em testes Gray-Box. (ALI e HERIYANTO, 2011, p. 39). Neste tipo de teste o auditor não conhece todos os detalhes da infraestrutura e/ou tecnologia envolvida.

8.4. Tandem

Nessa modalidade o auditor tem total conhecimento sobre o alvo, o alvo sabe que será atacado e o que será feito durante o ataque. Também conhecido como "Caixa de Cristal".

Esse tipo de pentest é bem próximo de uma auditoria, pois ambos estão preparados e sabem o que vai ser realizado. É o ideal para ser feito periodicamente, monitorando as vulnerabilidades novas e mudanças feitas na infraestrutura.

8.5. Reversal

Nessa modalidade o auditor tem conhecimento total do alvo, porém o alvo não sabe que será atacado, e tão pouco sabe quais testes serão executados.

Esse formato de teste é ideal para testar a capacidade de resposta e como está o timing de ação da equipe de resposta de incidente do alvo. (ALI e HERIYANTO, 2011 p.42).

9. White Hat ≠ Black Hat

No mundo do pen testing, não é incomun ouvir os termos "White hat" (chápeu branco) e Black Hat" (chapéu preto) é importante observar que os hackers éticos realizam das mesmas atividades usando as mesmas ferramentas que os invasores maliciosos. É importante observar que os hackers éticos realizam várias das mesmas atividades usando as mesmas ferramentas que os invasores maliciosos.

Em quase todas as situações, um hacker ético deve se esforçar para agir e para pensar como um hacker black hat de verdade. Quanto mais simulação do teste de invasão se assemelhar a um ataque do mundo real, mais valor isso terá para o cliente que estiver pagando pelo teste de invasão (penetration testing). Há um mundo de diferença entre os dois lados, essas diferenças podem ser reduzidas a três pontos principais: autorização, motivação e intenção. (ENGEBRETSON, 2014).

10. Kali Linux e o Backtrack: ferramentas, muitas ferramentas

Atualmente, um dos aspectos mais positivos em aprender a hackear está na abundância e na disponibilidade de boas ferramentas para realizar a sua arte. O Backtrack Linux e o Kali Linux são o sonho de qualquer pentester da área de segurança. Essas distribuições são totalmente voltadas para os pentesters. Ela já vem carregadas com centenas de ferramentas de segurança instaladas, configuradas e prontos para o uso. E o melhor de tudo é que o Kali e o Backtrack são gratuitos. (ENGEBRETSON, 2014).

11. Fases de um Teste de Invasão

Um ataque, ou teste de invasão, é composto por uma série de fases, onde em umas determinadas operações são realizadas. O que vai definir a diferença de um teste

de invasão e um ataque realizado por um cracker, são justamente a intenção, o escopo e o espaço de tempo disponível para o mesmo.

As fases básicas de um ataque são explicadas a seguir:

11.1 Levantamentos de Informações

Essa é a fase mais importante de um ataque e de um teste de invasão. Baseado no que é descoberto nessa fase, todo o planejamento é realizado e os vetores de ataque definidos. Essa fase prossegue na fase seguinte, onde as informações iniciais são estendidas, de forma mais detalhada. Podemos dizer que essa é a fase abrangente, e a fase seguinte detalha as informações adquiridas nessa primeira fase.

Qualquer informação que seja vinculado ao alvo é considerada de valor nesse primeiro passo:

- Concorrentes;
- Nome de funcionários;
- Endereços;
- Telefones;
- Sites;
- Empresas;
- Comunidades sociais;

11.2. Varredura

Nessa fase o atacante busca informações mais detalhadas do alvo, que posam permitir definir seus vetores de ataque e enxergar as possibilidades que podem permitir ganhar acesso ao sistema, através da exploração de alguma falha encontrada.

Aqui buscamos informações que respondam algumas perguntas, como por exemplo:

- Qual sistema operacional o alvo utiliza?
- Quais os serviços estão sendo executados no alvo?
- Quais serviços estão disponíveis para acesso?
- Qual a versão de cada serviço sendo executado?
- Há IDS/IPS na rede?
- Há firewalls na rede?
- Existe uma rede interna e outra externa, como uma DMZ?
- Há serviços com acesso público rodando em alguma máquina?

A partir dessas informações, o atacante pode buscar maiores detalhes na internet ou fóruns especializados em busca de exploits que permitam explorar falhas existentes nas versões dos serviços sendo executados. (RAMOS, 2008).

11.3. Ganhando Acesso

Aqui o atacante coloca em prática tudo aquilo que planejou a partir das informações obtidas previamente. Dependendo de seus vetores de ataque, ele pode realizar uma série de ataques buscando ganhar acesso ao sistema alvo, como por exemplo:

- Ataques de força bruta local;
- Captura de tráfego de rede;
- Ataque de engenharia social;
- Ataques às aplicações WEB;
- Exploração de sistema operacional

Conseguindo acesso ao sistema, o atacante realizará uma série de operações buscando a elevação de seus privilégios caso o mesmo já não seja de root.

11.4. Mantendo Acesso

Após conseguir o acesso, a atacante busca, de alguma forma, manter o acesso conseguido através de seus ataques. Isso normalmente não é utilizado por um pentester, a não ser que seja extremamente necessário. O risco de configurar o sistema, implantando backdoors ou outro tipo de dispositivo que permita o acesso posterior, é que a ferramenta utilizada pode voltar se contra você, pois outras pessoas podem descobri-la, explorá-la e ganhar acesso facilmente ao sistema comprometido.

Portanto, essa fase, quando realizada durante um teste de invasão, precisa de extremo cuidado e planejamento para não trazer comprometimentos e prejuízos desnecessários ao alvo.

11.5. Limpando os rastros

Nessa fase final do ataque, o atacante apaga todos os seus rastros, todos os registros de operações realizadas dentro do sistema comprometido. Como o pentester tem autorização para realizar os testes, não é necessário apagar rastros.

Isso se torna importante para um pentester, apenas se quiser testar, também, a capacidade da equipe de perícia forense e respostas a incidentes de descobrir o que foi feito e recuperar informações alteradas.

12. Metodologia existente

Para um teste de invasão não ficar “solto” e sem uma sequência lógico coerente, a comunidade de segurança, através de alguns órgãos, associações, institutos e pesquisadores, criou uma série de metodologias para servirem como guias básicos para a correta realização de testes de invasão. Isso permite uma certa padronização nos testes realizados seguindo uma outra metodologia. Podemos citar as seguintes metodologias conhecidas internacionalmente:

12.1. Open Source Security Testing Methodology Manual - OSSTMM

O OSSTMM é um padrão de segurança reconhecido internacionalmente. Trata-

se de um padrão puramente baseado em métodos científicos que consistem em quantificar a segurança operacional e seu custo de acordo com os objetivos do negócio.

12.2. Information Systems Security Assessment framework - ISSAF

O ISSAF é outro padrão aberto de análise e teste de segurança que foca em duas áreas, técnica e gerencial. Utiliza categorização em diversos domínios para endereçar os testes de segurança em uma ordem lógica. É uma metodologia que se encaixa perfeitamente no ciclo de vida do negócio em uma organização.

Com a pretensão de ser um framework bastante abrangente a mesma se baseia em informações atualizada de segurança, melhores práticas e considerações administrativas para complementar e programa de verificação de segurança. É um padrão que escolhe um caminho mais curto para alcançar seu deadline analisando seu alvo contra vulnerabilidades críticas que podem ser exploradas com mínimo de esforço. (ALI e HERIYANTO,2011, p.44).

12.3 Open Web Application Security Project – OWASP

Este framework engloba diversos projetos, um dos mais importantes é o Top 10. É um projeto que tem como objetivo educar desenvolvedores, designers, arquitetos de software, gerentes e organizações sobre as consequências das falhas de segurança mais críticas em uma aplicação web.

O projeto também provê técnicas básicas para proteção destas ameaças. Técnicas sofisticadas de proteção de perímetro e de host ainda sofrem para prevenir que uma aplicação segura envolve pessoas, processos, gerenciamento e tecnologia. O projeto aberto da comunidade chamado de OWASP é uma iniciativa que foca nos fundamentos para integrar segurança aos princípios de melhores práticas no desenvolvimento seguro.

Ataques e falhas desta visão são mapeados com o projeto OWASP Top Ten, Common Weakness enumeration (CWE) da Mitre, Common Attack Pattern Enumeration and classsication (CAPEC), também da Mitre e a lista SANS-CWE top 25. (ALI e HERIYANTO, 2011, p. 49).

13. Como conduzir um Teste de Invasão

Alguns passos básicos são necessários para a preparação e realização de um teste de invasão, para que o mesmo seja bem sucedido.

Suponha que você seja um pentester ético trabalhando para uma empresa de segurança. Seu chefe entra em seu escritório e entrega um pedaço de papel a você. “Acabei de conversar com a CEO daquela empresa por telefone. Ela quer que meu melhor funcionário – ou seja você – realize um pentest em sua empresa. Você acena com a cabeça, aceitando a tarefa. Ele sai. Você revira o papel – há apenas uma única palavra escrita: “Tabajara”. É uma empresa da qual você nunca tinha ouvido falar antes e não há nenhuma outra informação no papel. E agora? (ENGEBRETSON, 2014).

O primeiro passo em todo trabalho consiste em realizar uma pesquisa. Quanto mais você se preparar de forma completa para uma tarefa, maiores são as chances de ter sucesso.

Abraham Lincoln, que dizia: *“Se eu tivesse seis horas para derrubar uma*

árvore, eu gastaria a primeira delas afiando o meu machado."

14. Aspectos legais

É importante atentarmos para os aspectos legais de um teste de invasão, e se os mesmos estão de acordo com as leis vigentes no país, e principalmente com o que foi assinado no contrato de prestação de serviço.

Devemos lembrar-nos de uma coisa: TESTE DE INVASÃO SEM PERMISSÃO É CRIME!

Portanto, tenha sempre um contrato prévio assinado com o cliente, onde serão definidos os seguintes pontos:

- Limites do teste: até onde pode ir;
- Horários: períodos de menor utilização ou menos críticos;
- Equipe de suporte: caso haja alguém para tomar providências caso alguém ataque tenha efeitos colaterais;
- Contatos: ao menos três contatos, com e-mail, endereço e telefone;
- Permissão assinada: um documento assinado pelo responsável pela empresa, com os nomes das pessoas da equipe autorizadas a realizar os testes.

Dentro do que foi acordado, devemos ter o máximo cuidado para não causar comprometimentos que tragam algum tipo de prejuízo ao cliente, como a indisponibilidade de informações vitais para o funcionamento organizacional, por exemplo.

Levando em conta esse aspecto, se possível, é interessante reproduzir o ambiente de testes em máquina virtual para aproximar-se do possível comportamento do ambiente testado antes de finalmente lançarmos alguns tipos de ataques. (ARAUJO, 2012).

Isso evitaria a maior parte dos comprometimentos não planejados à

Infraestrutura do cliente, e pode poupar muita dor de cabeça!

15. Estratégia no Levantamento de Informações

Um levantamento ativo inclui a interação com o alvo. É importante observar que, durante esse processo, o alvo pode gravar nosso endereço IP e registrar nossas atividades em um log. Isso tem uma chance bem alta de ser detectado se estivermos tentando realizar um teste de invasão de maneira discreta.

O levantamento passivo faz uso da vasta quantidade de informações disponíveis na web. Quando conduzimos um levantamento passivo, não interagirmos diretamente com o alvo e, desse modo, ele não terá nenhuma maneira de saber, gravar ou registrar nossas atividades em um log.

16. HTTrack: clonador de sites

O HTTrack é um utilitário gratuito que cria uma cópia off-line idêntica do

site alvo. O site copiado inclui todas as páginas, links figuras e o código do site original, porém permanecerá em seu computador local.

Com o uso desta ferramenta nos permite explorar e minar completamente o site "off-line", sem a necessidade de tempo adicional para vasculhar o servidor web da empresa.

- # **httrack**
- - Usando o Kali, o site clonado ficará no diretório */root/websites/<nomedoprojeto>*
- - Ao abrir o Firefox inserir na URL */root/websites/<nomedoprojeto>*
- ***Obs:*** *<nomedoprojeto>* deve ser substituído pelo nome que foi usado ao criar a cópia.

17. Google Hacking

Google Hacking é a atividade de usar recursos de busca do site, visando atacar ou proteger melhor as informações de uma empresa. As informações disponíveis nos servidores web da empresa provavelmente estarão nas bases de dados do Google.

Um servidor mal configurado pode expor diversas informações da empresa no Google. Não é difícil conseguir acesso a arquivos de base de dados de sites através do Google. O Google possui diversos recursos que podem ser utilizados durante um teste de invasão, e justamente por isso é considerada a melhor ferramenta para os hackers, pois permite acesso a todo e qualquer tipo de informação que se queira. (LONG, 2005).

17.1. Comandos Avançados do Google

O Google é a principal ferramenta para o levantamento de informações de nosso alvo. É o melhor sistema público para utilizarmos em busca de informações sobre qualquer coisa em relação ao nosso alvo: sites, propagandas, parceiros, redes sociais, grupos e etc.

Para usar adequadamente uma diretiva do Google, três dados são necessários:

O nome da diretiva que você quer usar;

Dois pontos;

O termo que você quer usar com a diretiva.

Hands-on:

Localizar páginas potencialmente interessantes:

inurl:admin

inurl:login

Pesquisar por versão de cache da homepage:

cache: enucomp.com.br

Busca por arquivos de base de dados em sites do governo:

3. site:gov.br filetype:pptx
4. site:gov.br ext:SQL

Possíveis falhas em aplicações web:

5. allinurl:".php?site="

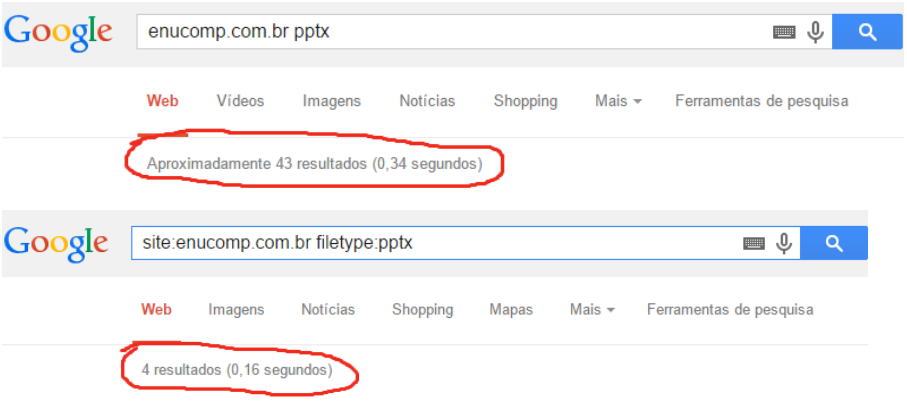


Figura 2. A eficiência das diretivas no Google

18. The Harvester: descobrindo e tirando proveito de endereços de e-mail

Uma excelente ferramenta a ser usada no levantamento de informações, criado por Christian Martorella da Edge Security. Essa ferramenta nos permite catalogar de forma rápida e precisa tanto os endereços de e-mail quanto os subdomínios diretamente relacionados ao nosso alvo. (ENGEBRETSON, 2014).

O The Harvester pode ser usado para pesquisas de e-mails, hosts e subdomínios em servidores Google, Bing e PGP, bem como por usuários no LinkedIn.

- # **theharvester ou # pentest/enumeration/theharvester**
- # **./theharvester.py –d <domínio> -l 10 –b <google, bing, Linkedin> all**

19. Extraindo informações do cabeçalho de Email

A análise de e-mails é normalmente feita pela área de Forense Computacional. Porém, podemos usar um e-mail para obter informações sobre o host da pessoa, quando esse não é de conhecimento do atacante e precisa ser descoberto, pois é um possível alvo.

O correio eletrônico é dividido em duas partes: Cabeçalho (Header) e Corpo (Body) do e-mail. No cabeçalho é onde encontramos diversos campos com informações de controle, destinatário, remetente, data de envio, dentre outras informações. E é no

corpo da mensagem que encontramos a mensagem, em si.

O campo que interessa a nós é o campo “Received:”, que contém informações sobre o endereço IP de onde a mensagem de correio eletrônico partiu.

```
X-Gm-Message-State:                                                    ALoCoQkfnN-
sE5EtTlI22aUBqKzB/DpkP0igOz7MNGBR7otlkvs6mk4895nYLfIOW4EIFIou8VB37

X-Received: by 10.112.91.41 with SMTP id cb9mr5444612lbb.53.1412703933687;

 Tue, 07 Oct 2014 10:45:33 -0700 (PDT)

MIME-Version: 1.0

Received: by 10.112.144.198 with HTTP; Tue, 7 Oct 2014 10:45:13 -0700 (PDT)

X-Originating-IP: [179.213.61.175]

From: Diane Malaquias <diane.malaquias@esecurity.com.br>

Date: Tue, 7 Oct 2014 14:45:13 -0300

Message-ID: <CAPEmNXaktJxPg2SdFLS037B1Z2UZFTyBkM9MtqSVUsnhn15zcg@mail.gmail.com>

Subject: =?UTF-8?Q?Analista_de_Seguran=C3=A7a_da_Informa=C3=A7=C3=A3o_PL?=

To: undisclosed-recipients:

Content-Type: multipart/alternative; boundary=001a11349fcc2401f80504d8c488

Bcc: neetfel@gmail.com
```

20. Levantamento de Informações

20.1. Footprint

Footprint é a etapa a ser realizada em um teste de intrusão. Durante essa etapa, o Pentester coleta o máximo de informações para alimentar a anatomia de ataque. Podemos dizer que é a fase em que o Pentester se prepara para realizar o ataque.

Em média, um Pentester gasta 70% do tempo analisando um alvo levantando informações sobre o mesmo. Apenas 30% do tempo e usado para realizar o ataque e avaliar a possibilidade de um atacante realizar procedimentos pós-invasão na máquina alvo.

Quando estamos realizando um footprint, devemos buscar informações relativas à topologia da rede, sistemas operacionais, quantidade de máquinas e localização física. Além disso, é importante também descobrir informações sobre os funcionários da empresa, como: e-mails, cargos e também função específica no ambiente.

20.2. Consulta a informações de domínio

Após observar o site do alvo, é de interesse do atacante conhecer detalhes referentes ao nome de domínio do cliente. A primeira coisa a ser feita é buscar informações sobre o proprietário de um domínio. Isso pode ser feito utilizando o comando whois.

- **# whois domínio.com.br**

Podemos concluir que com um simples comando, disponível em praticamente qualquer sistema operacional, foi possível obter o nome do responsável pelo domínio, o nome do responsável técnico pelo domínio, o nome dos dois servidores de DNS e o CNPJ da empresa, localizado no campo ownerid.

Além da consulta whois em sistemas operacionais, é possível ainda utilizar serviços que consultam a base de dados de proprietários de domínios através de ferramentas web, como o próprio http://registro.br, por exemplo.

- **https://registro.br/cgi-bin/whois/**

20.3. Consultando servidores DNS

Os servidores DNS são um alvo excelente para os hackers e os pentesters. Normalmente, esses servidores contêm informações consideradas altamente valiosas pelos invasores. O DNS é um componente essencial, tanto para as nossas redes locais quanto para a internet. Além de ser responsável por traduzir nomes de domínio em endereços IP.

Como pentesters, é importante focar nos servidores DNS que pertençam ao nosso alvo. Pois o mesmo deve conhecer tanto o endereço IP quanto o nome de domínio correspondente de cada computador em sua rede.

- **# nslookup**

20.4. Dig

Ótima ferramenta para extrair informações do DNS.

Vamos consultá-lo, então.

- **dig ip_alvo**
- **dig –t MX domínio.com.br**
- **dig –t NS domínio.com.br**

Os campos MX, e NS fornecem, respectivamente, o nome dos servidores de e-mail e o nome de todos os servidores de DNS.

Com essa consulta, já conseguimos, inclusive, endereços de servidores que utilizaremos em nossa varredura e enumeração de serviços.

Podemos tentar uma transferência de zona usando a ferramenta presente nas distribuições BackTrack e Kali usando o **dnsenum**

- **# /pentest/enumeration/dns/dsnenum**
- **#./dnsenum <domínio> dns.txt**

Se as transferências de zona forem permitidas e não estiverem restritas, você verá uma lista de hosts e de endereços IP do servidor DNS alvo relacionada ao seu domínio-alvo.

20.5. Consultando websites antigos

Além da possibilidade de utilizarmos a opção em cache do Google, é possível

utilizarmos outros serviços que possibilitam que acessemos versões mais antigas de qualquer site que já tenha sido publicado na web.

- **http://www.archive.org**

Com isso, podemos encontrar informações que podem ser úteis, principalmente para ataques de engenharia social, pois encontramos produtos antigos, ex-funcionários, informações que foram retiradas do site por serem sensíveis e etc.

20.6. Webspiders

Webspiders são programas que navegam automaticamente por websites para coletar informações. Pensando no Google, um webspider feito pelo Google navega pelos links das páginas e alimenta uma base de dados do Google, que é usada para consultas durante as buscas realizadas pelos usuários.

Um webspider consulta o arquivo robots.txt que está localizado no diretório raiz do website para saber quais arquivos ele não deve analisar. Portanto, os arquivos ou diretórios que estão listados em um arquivo robots.txt não aparecerão nos resultados das buscas realizadas em sites como o Google.

Vamos a um exemplo:

- **http://www.dominio.com.br/robots.txt**

Portanto, os arquivos robots.txt podem revelar para nós informações sobre arquivos e diretórios que poderíamos não conhecer e até mesmo não estar linkado no site.

20.7. Netcraft

Netcraft é uma empresa européia que prove serviços de internet. Dentro de alguns serviços que ela fornece está a análise de mercado para empresas de web hosting e servidores web, incluindo detecção do sistema operacional e versão do servidor web, e em alguns casos, informações sobre uptime do servidor, já que normalmente esse fator é determinante na escolha de uma empresa de hospedagem de sites.

Para nós, pode ser útil para exibir a versão do sistema operacional e servidor web que um determinado host está usando, além de manter um histórico das versões que o mesmo host já usou anteriormente.

- **http://www.netcraft.com**

21. O que é Engenharia Social?

Podemos considerar a engenharia social como a arte de enganar pessoas para conseguir informações, as quais não deviam ter acesso.

Muitas vezes empregados de uma empresa deixam escapar informações sigilosas através de um contato via telefone ou mesmo conversando em locais públicos como: corredores, elevadores e bares.

Uma empresa pode ter os melhores produtos de segurança que o dinheiro pode

proporcionar. Porém, o fator humano é, em geral, o ponto mais fraco da segurança. Não há dúvidas de que o levantamento de informações ou o hacking não seriam completos sem a inclusão da engenharia social. (MITNICK, 2003).

É o meio mais simples e eficiente para reunir informações sobre um alvo.

Figura 3. Ataque ao Service Desk

22. Tipos de Engenharia Social

22.1. Baseada em Pessoas

As técnicas de engenharia social baseada em pessoas possuem diversas características que são utilizadas para que o atacante consiga as informações que deseja, dentre elas podemos citar:

1. Disfarces;
2. Representações;
3. Uso de cargos de alto nível;
4. Ataques ao serviço de Helpdesk;

22.2. Baseada em Computadores

Esses ataques são caracterizados por utilizarem técnicas de ataque baseadas no desconhecimento do usuário com relação ao uso correto da informática.

- Cavalos de Tróia anexados a e-mails;
- E-mails falsos;
- WebSites falsos

23. Anonymizer

Os programas de anonymizer funcionam basicamente para ocultar seus dados enquanto navega na internet. Normalmente a aplicação utilizada para isso é um proxy,

que após configurado, permite que seu IP seja mascarado, fornecendo o dele como IP real.

Com isso, é possível proteger o conteúdo de e-mails, textos de softwares de mensagens instantâneas, IRC e outros aplicativos que usam o protocolo TCP. Uma boa ferramenta para utilizarmos mantendo nossos dados de navegação protegidos, é o

TOR – The Onion Router **http://www.torproject.org/**

24. Fingerprint

Fingerprint é uma das principais técnicas de levantamento de informação que é realizada por um Pentester antes que o mesmo comece a realizar os ataques em seu alvo.

A função dessa técnica é identificar a versão e distribuição do sistema operacional que irá receber a tentativa de intrusão. Sendo assim, essa técnica é extremamente importante para que o atacante consiga desenvolver de maneira mais precisa e menos ruidosa seu ataque. Usando essa técnica o Pentester estará explorando problemas da pilha TCP/IP e verificando características únicas que permitem que o sistema alvo seja identificado.

Para realizar tais análises, podemos utilizar ferramentas específicas, conhecidas como scanners de fingerprint, que são softwares usados para realizar tarefas de detecção de sistemas operacionais. (MELO, 2006).

Entre os scanners existentes, podemos dividi-los basicamente em dois tipos:

24.1. Fingerprint Passivo

O scanner atua como um "farejador" na rede, ou seja, fica escutando os pacotes que passam por ela, detectando o formato do pacote que está passando consegue identificar o sistema operacional.

Para esse tipo de operação, utilizamos a ferramenta p0f, que permite "farejarmos" os pacotes que trafegam na rede.

- # **p0f -i eth0 -o log**

Com o parâmetro -i definimos em qual dispositivo de rede ele ficará farejando os pacotes, se não definimos nada, ele assume "all", farejando todos os dispositivos disponíveis. Com o parâmetro -o, dizemos para o p0f armazenar tudo o que for capturado em um arquivo de saída, com nome definido por nós.

24.2. Fingerprint Ativo

O scanner envia pacotes manipulados e forjados, baseado em uma tabela própria de fingerprint. Com isso, ele analisa a resposta do pacote e compara com a tabela, para definir qual o sistema operacional.

O risco desse tipo de fingerprint, é que se o alvo estiver com um firewall bem configurado e um IDS/IPS, nosso acesso pode ser logado em seu sistema e pode ser difícil que consigamos muitas informações.

Duas ferramentas que podemos utilizar em um fingerprint ativo são nmap e

xprobe2, além, obviamente, dos comandos ping e traceroute, só para citarmos dois comandos básicos. (MELO, 2006).

25. Descobrindo um Sistema Operacional usando ICMP

Um simples ping é capaz de revelar o sistema operacional de uma máquina.

- # **ping <domínio> -c 1**

A informação importante está no campo TTL (Time To Live). A maioria dos sistemas operacionais se diferencia pelo valor retornado de TTL. Veja a lista abaixo:

1. Cyclades - Normalmente 30;
2. Linux - Normalmente 64
3. Windows - Normalmente 128
4. Cisco - Normalmente 255
5. Linux + iptables - Normalmente 255

26. Calculando o HOP

Utilizando os comandos traceroute e ping conjugados para obter informações, podemos calcular o ttl e descobrir o sistema operacional do alvo.

- # **traceroute <domínio>**

Com o traceroute podemos ver que temos 11 saltos, até que os pacotes são interrompidos, o que pode representar um firewall ou algo do tipo que descarte os pacotes.

Agora que sabemos por quantos roteadores estamos passando, podemos usar o comando ping para descobrir o TTL do site.

Somando a quantidade de saltos (11) com o valor de ttl (49), temos 60. O mais próximo de 60 é 64, que representa o Linux. A partir daí, podemos concluir que o sistema operacional utilizado no servidor onde o site está hospedado é Linux.

27. Fingerprint através do xprobe2

Ferramenta para fingerprint ativo apresentada na conferencia BlackHat LasVegas em 2001 criada por Fyodor criador da ferramenta nmap e Ofir Arkin cofundador do projeto honeynet.org.

Seu banco de dados de assinaturas fica em:

- **/usr/local/etc/xprobe2/xprobe2.conf**

Execute o xprobe2 na máquina do instrutor para descobrir o sistema operacional:

- **# xprobe2 <IP>**

Agora tente utilizar o fingerprint sobre uma porta aberta:

- **# xprobe2 -p tcp:80:open <IP>**

Percebe-se que quanto maior o número de informações que passamos para o xprobe2, maior é a precisão no reconhecimento do Sistema Operacional do alvo.

28. Enumeração de Informações e Serviços

As técnicas de enumeração são utilizadas como um complemento às fases de fingerprint e varredura. O objetivo é descobrir serviços e versões que estão sendo executados no sistema alvo, facilitando a posterior pesquisa de vulnerabilidades e exploits específicos. Quanto mais informações tivermos sobre nosso alvo, mais fácil será para encontrarmos vulnerabilidades e melhorarmos nossos vetores de ataque. Sabendo os serviços rodando e as versões dos mesmos, torna-se possível encontrarmos os exploits corretos e as vulnerabilidades que poderão ser exploradas.

Além disso, na fase de enumeração, mapeamento de toda a rede do alvo, descobrindo os pontos falhos e onde podemos explorar para conseguir acesso a informações estratégicas. (MELO, 2006).

29. Aquisição de Banners

Falaremos agora a respeito da captura de informações sobre os serviços que estão rodando em uma máquina-alvo por meio da leitura de seus respectivos banners que são aquelas mensagens que contém informações como o tipo do serviço, sua versão, etc. Essas informações visam estabelecer um levantamento dos serviços utilizados, onde o foco de um possível ataque pode estar voltado para a exploração de vulnerabilidades desses serviços.

29.1. Técnicas Clássicas

Sem utilizar ferramentas específicas, é possível conseguir informações dos serviços que estão sendo executados em determinada porta. Abaixo veremos dois exemplos, utilizando ftp e telnet.

Obtendo banner de um servidor ftp:

- **# ftp 192.168.200.254**

Obtendo banner de um servidor de e-mail:

- **# telnet 192.168.200.205 25**
- ***HELO [domínio]***
- ***MAIL FROM: [endereço_origem]***
- ***RCPT TO: [endereço_destino]***
- ***DATA***
- ***(... msg ...)***

- .
- *quit*

Com o comando telnet, podemos tentar conectar em todas as portas existentes para verificar o que está sendo executado. Obviamente que esse é um método lento e impossível de ser executado nas mais de 65000 portas existentes, mas é interessante conhecê-lo, e ainda evita a detecção por IDS ou firewall. (MELO, 2006).

30. Ferramentas

Nmap: Realiza varredura de rede, buscando hosts ativos, portas abertas e serviços sendo executados.

Xprobe2: Analisa banners de sistemas operacionais, comparando com um banco de dados interno, onde compara-os e informa o S.O. utilizado e a versão do mesmo.

Amap: Analiza banners de serviços que estão sendo executados, e informa o nome e versão.

AutoScan: Faz varredura na rede e informa hosts ativos, portas abertas e serviços sendo executados. Funciona através de uma interface gráfica.

Maltego: Faz varredura de redes, serviços, protocolos, domínios e várias outras opções, informando de forma gráfica a relação entres os hosts ativos.

Lanmap: Varre toda a rede e captura pacotes, criando ao longo de sua execução um arquivo .PNG com o mapa da rede, informando graficamente a relação das máquinas encontradas.

31. Prática dirigida

31.1. Capturando banner de aplicações (de forma ativa)

A partir da sintaxe abaixo, utilizando os programas Nmap, Xprobe2 e Amap, faça o reconhecimento dos serviços e sistemas operacionais rodando nas máquinas da rede alvo, e compare os resultados de cada programa.

- **nmap -sV -O [ip_alvo]**
- **Xprobe2 –p TCP:80:open <ip>**
- **Amap <ip> <porta>**

32. Mapeando graficamente a rede

Os programas de linha de comando funcionam muito bem para varrer redes, descobrir hosts ativos, serviços sendo executados e versões dos mesmos. No entanto, quando temos um mapa gráfico à mão, torna-se muito mais fácil visualizarmos a estrutura da rede e definirmos os vetores de ataque de forma mais consistente.

Algumas rodam em linha de comando, como o Lanmap, por exemplo, e outras possuem uma interface gráfica, que facilita a operação, como o Cheops e Maltego.

Lanmap

- **# lanmap -i eth0 -r 30 -T png -o /tmp/**

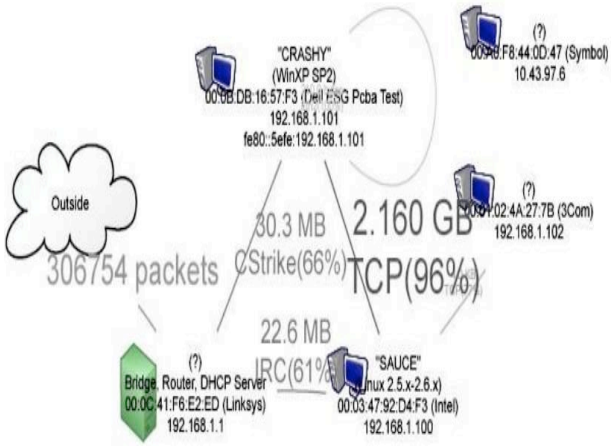


Figura 4. Mapeamento da Rede

AutoScan

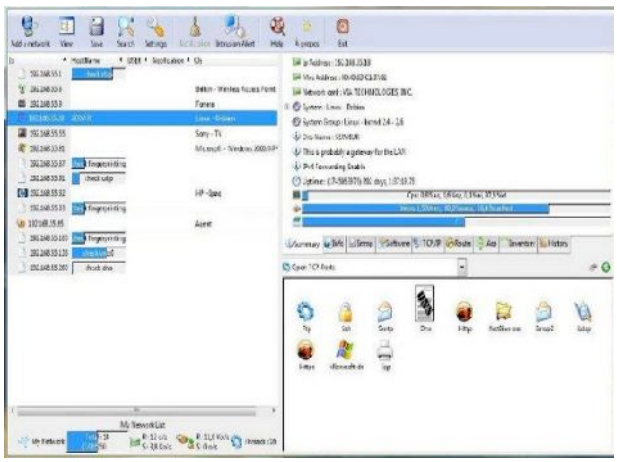


Figura 5. Scanning na Rede

Maltego

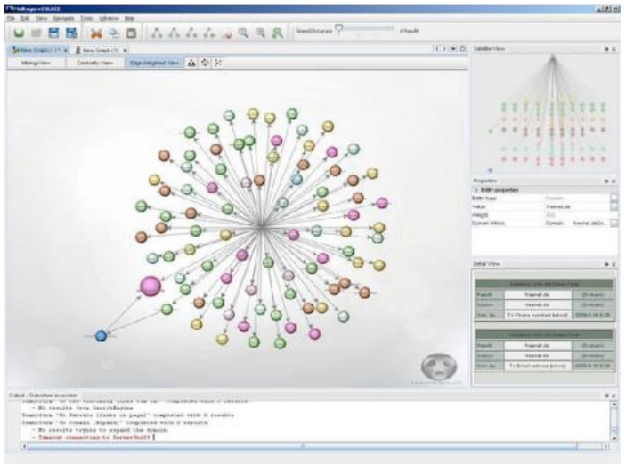


Figura 6. Reconhecimento

33. Vulnerabilidades em Aplicações Web

33.1 Entendendo a aplicação web

Aplicações web são programas que ficam em um servidor web e executam tarefas para dar uma resposta ao usuário. Uma aplicação web usa uma arquitetura cliente/servidor, normalmente com um navegador web como cliente e a web server como o servidor da aplicação.

Exemplos:

Webmails;

Web fóruns;

Blogs;

Lojas virtuais.

33.2 Porque é tão perigoso?

O objetivo de tentar explorar uma aplicação web é ganhar acesso a informações confidenciais. Aplicações web são críticas para a segurança de um sistema porque usualmente elas estão conectadas com uma base de dados que contém informações tais como cartões de crédito e senhas.

A maior parte dos ataques atualmente, não são realizados contra a infraestrutura organizacional, mas sim contra aplicações. E se houver falhas em aplicações WEB, muito possivelmente o atacante conseguirá acesso a todo conteúdo existente no servidor onde a aplicação está hospedada.

Na maioria das vezes, várias aplicações WEB ficam hospedadas em um mesmo servidor, compartilhando da mesma máquina física. Se uma, dessas várias aplicações hospedadas no servidor, tiver falhas graves, que permitam acesso à máquina, todas as outras serão comprometidas e o atacante também poderá explorar as demais máquinas acessíveis na rede. (PAULI, 2014).

34. SQL Injection

SQL Injection é um problema que ocorre quando o programa não filtra caracteres especiais enviados pelo usuário antes de fazer a requisição para o banco de dados, enviando caracteres que serão interpretados pelo banco de dados. SQL Injection é mais comum em aplicações web, porém outros tipos aplicações também podem estar vulneráveis.

Vamos analisar o trecho de código abaixo

- Select * from usuarios where username = ‘” + username + “’ and password
- = ‘” + password “’;

Como ficaria a chamada no banco de dados se enviássemos no username e password o conteúdo: ‘or ‘1’ = ‘1?

Reposta:

- Select * from usuarios where username = ‘’ or ‘1’ = ‘1’ and password = ‘’ or
- ‘1’=‘1’;

Onde:

- ' - Fecha a variavel de entrada do usuario
- OR - Continua a expressão SQL
- 1=1 - Uma expressão verdadeira

Como 1 é sempre igual a 1, teremos uma “verdade” e passaremos pela checagem. Esse é um tipo de dados que poderíamos passar para aplicativos vulneráveis e burlar o sistema de autenticação.

Exemplos de SQL Injection:

- **' or '1**
- **' or '1'='1**
- **' or 1=1-**
- **'or''='**
- **' or 'a'='a**

- **') or ('a'='a**
- **'or '=1**

34.1. Ferramentas utilizadas:

Windows	**Linux**
Pangolin	SQLmap
Havij	

35. Testando o Sistema

35.1. O que é negação de serviço?

Durante um ataque de negação de serviço, o atacante deixa o sistema impossível de ser usado ou significantemente lento, a ponto de conseguir realizar poucas tarefas. Esses ataques podem ser realizados contra um sistema individual ou contra uma rede inteira e normalmente são realizados com sucesso.

Qualquer tipo de ataque que afete o pilar "Disponibilidade" da tríade Confidencialidade-Integridade-Disponibilidade, pode ser considerado um ataque de negação de serviço, desde puxar a tomada de alimentação de energia de um servidor, até utilizar uma rede zumbi para ataque em massa.

Na maior parte das vezes, o objetivo do atacante não conseguir acesso à informação, roubo de dados, ou tomar o controle da máquina. O objetivo é realmente causar a indisponibilidade de serviços nos sistemas do alvo, e isso pode levar a potenciais prejuízos financeiros, do ponto de vista comercial, por exemplo. (MELO, 2006).

35.2. DoS

De acordo com a definição do CERT (Computer Emergency Response Team), os ataques DoS (Denial of Service), também denominados Ataques de Negação de Serviços, consistem em tentativas de impedir usuários legítimos de utilizarem um determinado serviço de um computador.

Para isso, são usadas técnicas que podem: sobrecarregar uma rede a tal ponto em que os verdadeiros usuários dela não consigam usá-la; derrubar uma conexão entre dois ou mais computadores; fazer tantas requisições a um site até que este não consiga mais ser acessado; negar acesso a um sistema ou a determinados usuários.

É importante frisar que quando um computador/site sofre ataque DoS, ele não é invadido, mas sim, tem apenas o serviço parado. Os ataques do tipo DoS mais comuns podem ser feitos devido a algumas características do protocolo TCP/IP (Transmission Control Protocol / Internet Protocol), sendo possível ocorrer em qualquer computador que o utilize.

Uma das formas de ataque mais conhecidas é a SYN Flooding, onde um computador tenta estabelecer uma conexão com um servidor através de um sinal do TCP conhecido por SYN (Synchronize). Se o servidor atender ao pedido de conexão,

enviará ao computador solicitante um sinal chamado ACK (Acknowledgement).

O problema é que em ataques desse tipo, o servidor não consegue responder a todas as solicitações e então passa a recusar novos pedidos. (MELO, 2006).

Não são apenas grandes quantidades de pacotes geradas que podem causar um ataque de negação de serviço. Problemas em aplicativos também podem gerar.

- **Ping da morte**

Envia pacotes de tamanho elevado e numa frequência também alta (milhares de vezes por segundo).

- **Ping –i 1 –l 65500 (ip de destino ou nome host) –t**
- **-i 1** – o intervalo entre cada ping. No caso, 1 ms.
- **-l** 65500 – o tamanho do pacote, em bytes **(este é o maior tamanho possível)**.
- **Alvo – o IP ou o nome** (se houver uma tabela de hosts ou um servidor DNS disponível) do destino.
- **-t** enviar a requisição por tempo indeterminado ou até conhece cancelar (CONTROL + C)
-

35.3. Exemplos de DoS

Fork Bomb :(){ :|:& };:

- Cria uma função
- função (){
- Dentro dela chama ela mesma e direciona a saída pra ela mesma “Looping”
- função | função &
- E depois chama a função
- }; função

35.4. Ferramentas DoS

O C4 é uma ferramenta que gera ataques de DoS em redes locais com SYN Flood Vamos conhece-la um pouco mais, digitando no terminal:

- $./c4

Com esse comando, teremos como retorno a sintaxe e a explicação resumida dos

parâmetros e opções do comando c4. A sintaxe correta para um ataque de SYN Flood com o c4 contra um host específico é:

Sintaxe: ./c4 –h [ip_alvo]

E alguns dos parâmetros existentes são:

- -h destination ip/host;
- -p destination port range [start,end] (defaults to random);
- -t attack timeout (defaults to forever);
- -l % of box link to use (defaults to 100%).

36. DdoS Distributed Denial of Service

O DdoS é um ataque DoS ampliado, ou seja, que utiliza até milhares de computadores para atacar um determinado alvo.

Esse é um dos tipos mais eficazes de ataques e já prejudicou sites conhecidos, tais como os da CNN, Amazon, Yahoo, Microsoft e eBay. Para que os ataques do tipo DdoS sejam bem-sucedidos, é necessário que se tenha um número grande de computadores para fazerem parte do ataque. Uma das melhores formas encontradas para se ter tantas máquinas, foi inserir programas de ataque DDoS em vírus ou em softwares maliciosos.

Em um primeiro momento, os hackers que criavam ataques DDoS tentavam "escravizar" computadores que agiam como servidores na internet. Com o aumento na velocidade de acesso à internet, passou-se a existir interesse nos computadores dos usuários comuns com acesso banda larga, já que estes representam um número muito grande de máquinas na internet.

Para atingir a massa, isto é, a enorme quantidade de computadores conectados à internet, vírus foram e são criados com a intenção de disseminar pequenos programas para ataques DoS. Assim, quando um vírus com tal poder contamina um computador, este fica disponível para fazer parte de um ataque DDoS e o usuário dificilmente fica sabendo que sua máquina está sendo utilizado para tais fins. (MELO, 2006).

Como a quantidade de computadores que participam do ataque é grande, é praticamente impossível saber exatamente qual é a máquina principal do ataque. Quando o computador de um internauta comum é infectado com um vírus com funções para ataques DoS, este computador passa a ser chamado de zumbi.

Após a contaminação, os zumbis entram em contato com máquinas chamadas de mestres, que por sua vez recebem orientações (quando, em qual site/computador, tipo de ataque, entre outros) de um computador chamado atacante. Após receberem as ordens, os computadores mestres as repassam aos computadores zumbis, que efetivamente executam o ataque. (MELO, 2006).

Um computador mestre pode ter sob sua responsabilidade até milhares de computadores. Repare que nestes casos, as tarefas de ataque DDoS são distribuídas a um "exército" de máquinas escravizadas.

Daí é que surgiu o nome Distributed Denial of Service. A imagem abaixo ilustra a hierarquia de computadores usadas em ataques DDoS. (MELO, 2006).

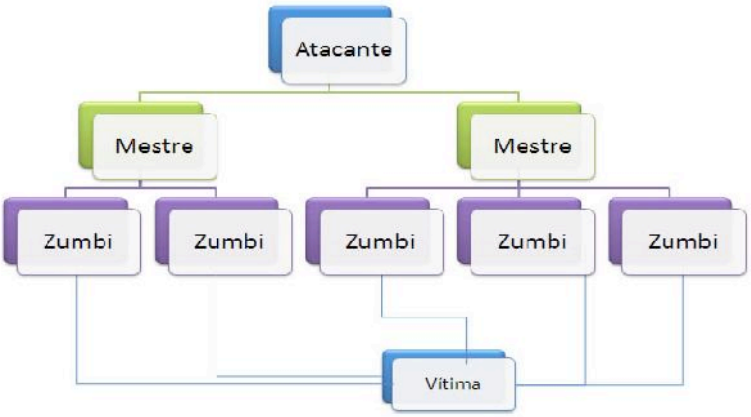


Figura 7. Ataque DDoS

36.1. Tipos de ataques DDoS

Ping Flood: É um ataque de negação de serviço simples no qual o atacante o sistema vítima com pacotes ICMP Echo Request (pacotes ping).

Este ataque apenas é bem sucedido se o atacante possui mais largura de banda que a vítima. Como a vítima tentará responder aos pedidos, irá consumir a sua largura de banda impossibilitando-a responder a pedidos de outros utilizadores.

As únicas maneiras de proteger deste tipo de ataque é limitando o tráfego do ping na sua totalidade ou apenas limitando o tráfego de pacotes ICMP Echo Request com um tamanho menor elevado.

- **ping –i 0 –s 65000 HOST**

36.2. Ferramentas DDoS

Loic: O site alvo é inundado com pacotes de requisição TCP ou UDP com a intenção de sobrecarregar o servidor, fazendo com que ele deixe de responder às requisições legítimas.

É frequente o uso de botnets para efetuar ataques através do LOIC.

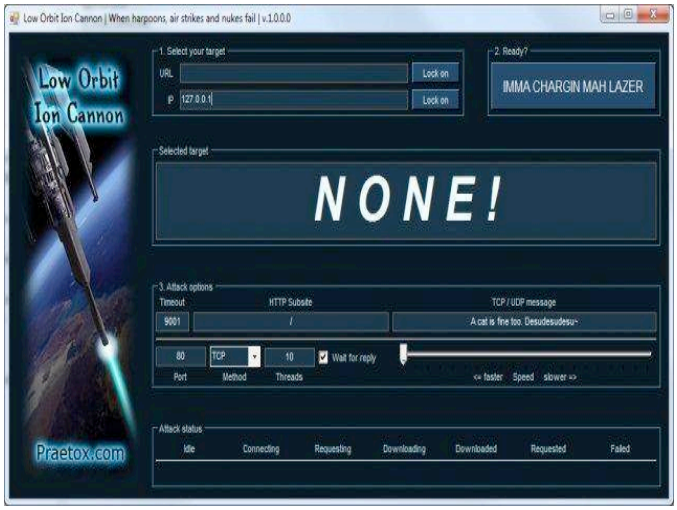


Figura 8. Loic

SlowLoris

- **# perl –MCPAN –e 'install IO::Socket::INET'**
- **# perl –MCPAN –e 'install IO::Socket::SSL'**

Funcionamento:

Tentar manter muitas conexões aberta com servidor web destino e mantê-los abertos o maior tempo possível até encher o máximo de conexões simultânea.

- **# perl slowloris.pl –dns <domínio>**

t50

Funcionamento:

Envia um número altíssimo de requisições de pacotes, de tal forma que o alvo não consiga atender a todas requisições ou as atenda de forma lenta, dessa forma o alvo pode cair ou ficar lento.

- **# ./t50 <dominio> --turbo --syn --flood**

O t50 é capaz de emitir as seguintes requisições:

- Mais de 1.000.000 (1 milhão) de pacotes por segundo de SYN Flood (+50% do uplink da rede) em uma rede 1000BASE-T (Gigabit Ethernet).
- Mais de 120.000 pacotes por segundo de SYN Flood (+60% do uplink da rede) em uma rede 100BASE-TX (Fast Ethernet).

3.37. CONCLUSÃO

A metodologia, técnicas e ferramentas foram idealizadas para suprir as necessidades dos novos administradores. No passado, era comum instalarmos um servidor e simplesmente esquecemos dele, já que o acesso era exclusivo da LAN da corporação. Com advento da internet tudo mudou: uma vez na internet, seu servidor está ao alcance do mundo.

Após ter concluído os detalhes técnicos de um teste de invasão, ainda resta uma tarefa. É necessário sintetizar suas descobertas na forma de um relatório de teste de invasão. Não é incomum encontrar hackers e pentesters extremamente habilidosos que queiram ignorar completamente essa atividade final.

O relatório de teste de invasão normalmente representa a imagem de sua empresa e sua reputação. É importante disponibilizar os dados detalhados de saída de cada uma de suas ferramentas. Poucos clientes terão a capacidade de entender os dados técnicos de saída.

Uma das melhores características de um teste de invasão e de um Hacking é que nunca se chega ao final. Assim que você domina um tópico ou uma técnica em particular, alguém desenvolve um método, um ataque ou um procedimento novo.

Isso não quer dizer que o projeto original de habilidades se tornará obsoleto. Pelo contrário, uma sólida compreensão do básico proporcionará uma base duradoura para aprender sobre tópicos mais avançados e permanecer em dia com ritmo rápido das mudanças.

Referências

ALI, Shakeel; HERIYANTO, Tedi. BackTrack 4: Assuring Security by Penetration Testing. 1ª Edição. Birmingham: Pack Publishing, 2011.

ARAUJO, Eduardo Edson de. A vulnerabilidade humana na segurança da informação. 2005. Disponível em <http://svn2.assembla.com/svn/GSIDEI/Bibliografia/Monografia%20Interven%C3%A7%C3%A3o%20humana%20Seguran%C3%A7a.pdf>. Acesso em: 01 de Outubro de 2012.

ENGEBRETSON, Patrick. Introdução ao Hacking e aos Testes de Invasão: Novatec, 2014.

MELO, Sandro. Exploração de Vulnerabilidades em Redes TCP/IP: Alta Books, 2006 2ª Edição atualizada.

MITNICK, Kevin D. A arte de enganar. São Paulo: Pearson Education do Brasil Ltda, 2003.

MITNICK, Kevin D.; SIMON, Willian L. A arte de invadir. São Paulo: Pearson Education do Brasil Ltda 2006.

PAULI, Josh. Introdução ao Web Hacking: Novatec, 2014.

RED HAT. CentOS Project. Disponível em: http://www.centos.org/docs/5/html/Deployment_Guide-enUS/ch-sec-access.html. Acesso em 11/10/2014.

SEMOLA, Marcos. Gestão da Segurança da Informação**:** uma visão executiva. Rio de Janeiro: Campus, 2002.

TI EXAMES. Curso preparatório para o exame ISO/IEC 27002 FOUDATION**,** 2011.

Um minicurso para inspeção das heurísticas de usabilidade para dispositivos móveis: refletindo sobre a qualidade dos apps

Samira e Silva Amaral Ribeiro[1]
Bruno Andrade da Silva[2]
Adriano Bessa[1]
Elizabeth Furtado[1]

Resumo: A quantidade de aplicações desenvolvidas para dispositivos móveis vem demonstrando um crescimento constante e a questão da usabilidade desses aplicativos nem sempre recebe a devida importância pela equipe de desenvolvimento, por conseguinte, encontramos no mercado muitos aplicativos em que sua utilização torna-se uma tarefa frustrante aos usuários. Os motivos podem ser vários, seja pelos usuários não conseguirem interagir com o aplicativo, seja pela dificuldade deles realizarem tarefas básicas que atendam suas expectativas. A inspeção de usabilidade é uma das maneiras de identificar problemas de usabilidade a partir de heurísticas e tem como objetivo melhorar a qualidade de sistemas quando se dispõe de pouco recurso. A literatura já apresenta registros de heurísticas específicas para aplicações de celulares segundo a revisão literária realizada neste trabalho. Assim, apresentamos um minicurso para que desenvolvedores e líderes de equipes possam aplicar inspeção com heurísticas de usabilidade que foram classificadas e organizadas com foco nas aplicações para dispositivos móveis.

Palavras-chave: aplicativos; inspeção de heurística; minicurso; qualidade; usabilidade.

Abstract: *The amount of applications developed for mobile devices has demonstrated steady growth and the question of the usability of these applications is not always given due importance by the development team therefore found many applications in the market as their use becomes a frustrating task to users. The reasons could be several, is that users can not interact with the application, or by their difficulty performing basic tasks that meet their expectations. The usability inspection is one of the ways to identify usability problems from heuristics and aims to improve the quality of systems when it has little recourse. The literature already presents records of specific heuristics for mobile applications according to the literature review conducted in this work. Thus, we present a short course for developers and team leaders can apply inspection with usability heuristics that have been classified and organized with a focus on mobile applications*.

Keywords: *applications; heuristic inspection; quality; usability, workshop.*

[1] Programa de Pós-graduação e Informática Aplicada – Unifor – Fortaleza – CE
samirarb@gmail.com, {adrianoba, elizabet@unifor.br}

[2] Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA – Altamira – PA
bruno.andrade@ifpa.edu.br

1. Introdução

A popularização dos smartphones e seu uso por pessoas de diferentes idades, gêneros, culturas, personalidades e habilidades cognitivas, torna-se um grande desafio aos desenvolvedores de aplicativos móveis que têm a responsabilidade de atender minimamente uma variedade de preferências, experiências e maneiras de interagir com tecnologias. Esta questão torna-se fundamental para o sucesso ou insucesso desses aplicativos, visto que, uma pessoa ao utilizá-los sem atingir as suas expectativas pode nunca mais voltar a usá-los, gerando um marketing negativo em seu network. Contudo, uma experiência bem sucedida pode tornar o usuário um consumidor fiel do aplicativo, o que algumas vezes pode torná-lo mais resistente em trocá-lo por outro, dada a sua segurança e bem estar com o atual aplicativo.

Neste contexto, a diversidade de perfis de usuários tem um impacto direto no processo de desenvolver uma interface intuitiva. Além deste fator, a qualidade da experiência do usuário depende da usabilidade dos aplicativos móveis.

Existem diversas atividades para identificar problemas relacionadas à usabilidade dos aplicativos, a avaliação é uma delas, a qual é de fácil aplicação e de baixo custo para uma organização e/ou profissional.

Consultando a literatura de Interação Humano-Computador (IHC) são retornadas muitas referências que abordam este assunto, entretanto a grande maioria dos trabalhos está relacionada às heurísticas de usabilidade para a web, com destaque para as heurísticas de Jakob Nielsen, fundador do movimento “discout usability engineering” que enfatiza métodos rápidos e eficientes para melhorar a qualidade de interfaces com o usuário. Nielsen é conhecido como o principal especialista mundial em usabilidade na web pelo U.S. News and World Report.

Quando o cenário é modificado para dispositivos móveis, existem poucas ocorrências, mas com o crescente uso dos consumidores, percebe-se que este ramo está em pleno crescimento e as pesquisas estão em andamento.

O presente artigo está dividido da seguinte forma: a seção 1 apresenta uma introdução; a seção 2 descreve sobre aplicações móveis; a seção 3 apresenta conceitos de avaliações de usabilidade; a seção 4 apresenta tabelas de uma revisão literária de uma pesquisa em andamento; a seção 5 descreve o minicurso com seus objetivos, ementa, carga horária e metodologia; a seção 6 relata experiências anteriores e um projeto piloto; na seção 7 conclui-se o trabalho descrevendo os trabalhos futuros e por fim na seção 8 registra-se os devidos agradecimentos.

2. Aplicações móveis

Oliveira et al. [1], afirmam que a inclinação ao uso de smartphones e tablets ocorre, principalmente, pela gama de aplicativos disponíveis em cada sistema. Os smartphones possuem aplicativos listados em diversas categorias como, entretenimento, medicina, finanças, e disponibilizados pelas lojas de aplicativos online de cada sistema operacional móvel. Como mostrado em [2], as aplicações para dispositivos móveis são, em geral, relativamente pequenas, feitas por um ou dois desenvolvedores e classificadas entre aplicações nativas e aplicações web. Também é observado que os desenvolvedores seguem um conjunto de “boas práticas” descritas em fóruns e comunidades de discussão, mas não um processo formal de prototipação, desenvolvimento e testes.

Esta falta de planejamento e a pouca documentação afetam diretamente a qualidade do aplicativo. Entende-se que o projeto de interface de um aplicativo deve ter como foco a usabilidade desse aplicativo, desta forma, propõe-se neste trabalho um minicurso de 8 horas para aplicar inspeção de heurísticas junto aos stakeholders, com isso, melhorar a percepção destes do impacto gerado pela falta de usabilidade nos aplicativos móveis.

3. Avaliações de usabilidade

Segundo Rocha e Baranauskas [3], uma avaliação de usabilidade tem três grandes objetivos: avaliar a funcionalidade do sistema (se está adequada aos requisitos da tarefa do usuário), avaliar o efeito da interface junto ao usuário (avaliar sua usabilidade) e identificar problemas específicos do sistema (aspectos do design os quais quando usados no contexto alvo, causam resultados inesperados ou confusão entre os usuários). Os métodos comumente adotados para avaliação de usabilidade podem ser divididos em duas grandes categorias [4]: (1) Inspeções de Usabilidade, nas quais inspetores examinam aspectos da aplicação para detectar violações de princípios de usabilidade estabelecidos; e (2) Testes de Usabilidade, que são métodos de avaliação baseados na participação direta de usuários. Embora o teste de usabilidade seja considerado o método mais eficaz para avaliar sistemas e protótipos do ponto de vista do usuário das aplicações, seu custo é alto, pois envolve o tempo dos usuários e algumas vezes o uso de laboratórios específicos de usabilidade [5]. Os métodos de inspeção foram propostos como uma alternativa para custo-benefício em comparação com os testes de usabilidade.

4. Revisão da literatura

Consiste em uma metodologia de pesquisa que tem como finalidade apresentar uma avaliação sobre um tópico de pesquisa. Foi realizada uma revisão da literatura tendo como foco heurísticas de usabilidade desenvolvidas para dispositivos móveis. Nesta pesquisa obteve-se um total de 220 artigos (Tabela 1) das bases IEEE (Institute of Electrical and Electronics Engineers), ACM (Association for Computing Machinery) e SCOPUS.

Tabela 1: Quantidade de artigos retornados nas bases

Bases Digitais	ACM	IEEE	SCOPUS
Quantidade de Artigos	23	72	125

Diante desta quantidade de trabalhos retornados, foram criados critérios de exclusão (E) para o refinamento da pesquisa, a saber:

- E1 – Artigos com idiomas diferentes do Inglês e do Português;
- E2 – Artigos de mesmo tema e autor;
- E3 – Disponibilidade apenas de título e resumo;
- E4 – Trabalhos onde o uso de heurísticas não esteja relacionado a usabilidade de software;
- E5 – Textos que não apresentam exemplos ou estudos que contribuam para identificação e customização de heurísticas de usabilidade para celulares.

Após aplicar estes critérios, obteve-se 7 (sete) artigos e neles foi identificado a existência de heurísticas adaptadas a partir das heurísticas de Nielsen para celulares. Assim, consideramos aplica-las no minicurso como insumo da atividade de inspeção, com o objetivo de melhorar a usabilidade e consequentemente a qualidade dos aplicativos móveis.

Na Tabela 2, representamos os trabalhos retornados e selecionados após critérios de exclusão, organizados pelas heurísticas de Nielsen que serão adaptadas para avaliação de aplicativos móveis.

Tabela 2: Interpretação de heurísticas de Nielsen para telefones celulares

Heurísticas de Nielsen	Interpretação (adaptação para celulares)
1) Visibilidade do status do sistema	[6] Deve fornecer em cada tela, informação clara e concisa sobre o status do sistema, devido ao tamanho de tela limitado e o grau de atenção durante a interação. [7] Deve possuir feedback facilmente interpretado. [8] Deve manter o usuário informado sobre todos os processos e estados do sistema através de um feedback e em um tempo razoável.
2) Concordância do sistema com o mundo real	[7] Deve utilizar a linguagem do usuário e possuir objetividade e clareza da mensagem. [8] Deve utilizar a linguagem do usuário, em vez de usar conceitos técnicos. Seguir as convenções do o mundo real e exibir a informação de forma lógica e na ordem natural.
3) Controle e liberdade do usuário	[6] As interrupções devem ser tratadas naturalmente pelo sistema, e este deve oferecer ao usuário a possibilidade de reiniciar imediatamente as suas ações. [7] Deve haver recuperação do estado anterior. [8] Deve permitir que o usuário desfaça e refaça as suas ações e forneça claramente "saídas de emergência" para sair dos estados indesejados.
4) Consistência e padrões	[6] Interface móvel deve ser semelhante às interfaces de desktop em termos de botões, logotipos e esquemas de cores para que a experiência do usuário seja consistente em todas as plataformas. [7] Deve manter os componentes no mesmo lugar e ao longo da interação, facilitar o aprendizado e estimular a memória de curto prazo do usuário. Funcionalidades semelhantes devem ser realizadas por interações semelhantes. A metáfora de cada componente ou recurso. [8] Deve seguir as convenções estabelecidas, de tal forma que o usuário seja capaz de executar as ações de forma "familiar", de maneira padronizada e consistente.
5) Prevenção de erros	[6] Deve usar procedimentos complexos para confirmar ações arriscadas a fim de evitar erros acidentais. Exemplo:

	Uso de controle de slide-to-unlock usado por Android e Apple, bem como controle de slides, da Apple, que controla ações, tais como, responder, desligar e excluir. [7] O usuário não deve ser forçado a memorizar informações ao passar de uma parte do diálogo para outra. A interface deve minimizar o esforço gasto para executar uma tarefa, exigindo pouca necessidade de memorização. Quanto menos memória for exigida melhor a aceitação. [8] Deve ocultar ou desativar funcionalidades indisponíveis, avisar os usuários sobre ações críticas e fornecer acesso a informações adicionais.
6) Reconhecer ao invés de lembrar	[7] Deve ser fácil o acesso às funcionalidades; Quanto a minimização da carga de memória do usuário, não deve ser necessário lembrar-se de informações a partir de uma tela para outra para completar uma tarefa. A informação da interface deve ser clara e suficiente para o usuário completar a tarefa atual. "Reconhecer ao invés de lembrar ": Usar o espaço disponível na tela; Tornar visíveis as informações apresentadas na tela; E facilitar a entrada de dados, são categorias de erro que os autores julgaram está relacionadas a esta heurística. [8] Deve apresentar de forma visível os objetos, ações e opções a fim de evitar que os usuários tenham que memorizar informações entre a execução de uma operação e outra.
7) Flexibilidade e eficiência de uso	[6] Deve possuir cobrança rápida do sistema e facilidade de aprendizagem, durante a interação com os telefones móveis, que normalmente dura menos do que a interação com desktops. Existem diversas maneiras de exibir a mesma função, ex.: atalhos podem ser adicionados, aumentando a eficiência da interação. Com base em dados do usuário e do contexto, o sistema deve, sempre que possível, sugerir apoio e opções de personalização para as ações frequentes. [7] Deve possuir personalização de ações frequentes. [8] Chamada de "Personalização e atalhos" pelo os autores, esta heurística deve fornecer opções de configuração básica e avançada, permitindo a definição e personalização do sistema. Além disso fornecer atalhos para ações frequentes.
8) Estética e designer minimalista	[6] Deve ser consistente, dividida em pequenas porções e fáceis de encontrar, a fim de minimizar o carga cognitiva do usuário. Além disso, para informações mais completas, o sistema deve permitir o acesso a versões desktop do site a partir de um telefone celular; A área da tela não deve ser coberta por um teclado; A informação mais relevante deve ser destacado visualmente através de um tamanho maior, cor, uso de marcadores, etc; E devido ao limite de tamanho da tela, ao contrário de sites para desktop, a navegação não deve ser repetida em todas as páginas. [7] Deve possuir um bom posicionamento dos componentes de interface, podem contribuir com a estéticas de aplicações móveis. [8] Deve evitar de informações indesejadas num contexto definido de utilização.

9) Ajudar os usuários a reconhecer, diagnosticar e corrigir erros	[6] Fornecer feedback significativo, mensagens de erro concisas, opções de retorno e recuperação de erro fácil, devido à entrada de dados limitada e consequentes maiores taxas de erros do usuário. [7] Deve prever erros e recuperar o estado anterior do sistema. [8] Deve exibir mensagens de erro em uma linguagem familiar para o usuário, informando sobre o problema e sugerindo uma solução construtiva.
10) Ajuda e documentação	[7] Deve ter uma opção de ajuda onde os problemas comuns e as formas de resolvê-los são especificados. As questões consideradas nesta opção deve ser fácil de encontrar. [8] Deve ser fácil de encontrar documentação e ajuda no dispositivo. Esta documentação e ajuda, deve está centrada na ação do usuário e deve indicar medidas concretas a seguir.

5. Minicurso

O minicurso intitulado "Inspeção das heurísticas de usabilidade para dispositivos móveis: refletindo sobre a qualidade dos apps" é uma proposta de formação inicial e continuada para desenvolvedores e líderes de equipe de desenvolvimento para aumentarem a usabilidade e a qualidade de suas aplicações móveis a partir da inspeção. Teve como insumo para sua idealização, a revisão literária descrita anteriormente.

Foi realizado um projeto piloto no mês de outubro na Universidade de Fortaleza com a turma de Graduação em Ciência da Computação e a participação de alunos da disciplina de Projeto de Interface. Esse projeto piloto ocorreu no laboratório de estudos do usuário e de qualidade do uso de sistemas (LUQS), situado nessa universidade, onde atuam pesquisadores da área de Interação Humano Computador (IHC) e os resultados serão descritos na próxima seção.

5.1. Objetivos

Oferecer condições teóricas e práticas, mínimas, que permitam o(a) participante reconhecer a importância da relação: Inspeção de Usabilidade x Qualidade do Aplicativo, a partir das Heurísticas de Nielsen e suas adaptações para celulares ou dispositivos móveis.

5.2. Ementa e carga horária

O ementário do minicurso foi dividido em: conceitos de qualidade de software; conceitos de inspeção de usabilidade; heurísticas de Nielsen; heurísticas de usabilidade para celulares; adaptação para dispositivos móveis; inspeção web na prática; inspeção em dispositivos móveis na prática; relatório de inspeção: métricas e indicadores. Como carga horária mínima do curso, foi idealizado 8 (oito) horas, conforme metodologia descrita na seção 5.3, contudo, o contexto (perfil dos stakeholders e da organização) irá determinar a carga horária ideal para este minicurso.

5.3. Metodologia

O minicurso prevê dois palestrantes que atuarão como facilitadores e agentes de construção do conhecimento dos participantes. Está dividido em quatro etapas e tem a duração de 8 horas.

Na primeira etapa ocorrerá uma apresentação pessoal dos palestrantes e suas respectivas pesquisas e interesse por meio de uma aula expositiva, na sequência haverá a apresentação de vídeo de curta duração, sobre Qualidade de Software, aplicação de questionário e socialização das respostas para nivelamento de conhecimento.

A segunda etapa inicia com uma aula expositiva dialogada com apresentação de Slides sobre Inspeção de Usabilidade e Heurísticas, finalizando a parte teórica do curso.

Na terceira etapa os participantes serão organizados em Grupos (formados aleatoriamente) para executar na prática uma inspeção de usabilidade na web, esta pratica será dividida em duas fases, na 1ª fase a inspeção será de uma única tela para todos os Grupos e na 2ª fase, cada grupo terá um conjunto de telas distintas. Haverá uma rápida socialização da inspeção e um feedback dos palestrantes à cerca dos acertos de cada Grupo.

Na quarta etapa cada membro do Grupo formado irá compor um Novo Grupo, na chamada dinâmica de mosaico integrado e realizarão inspeção de usabilidade em aplicações móveis, na 1ª fase a inspeção será de uma única tela para todos os Grupos e na 2ª fase, cada grupo terá um conjunto de telas distintas. Haverá uma rápida socialização da inspeção e um feedback dos palestrantes à cerca dos acertos de cada Grupo.

Para encerrar o minicurso os palestrantes apresentarão slides com considerações finais e realizarão auto avaliação do minicurso. Além disso, será aplicado um questionário para identificação do perfil dos participantes e seus respectivos níveis de conhecimento sobre heurísticas de usabilidade.

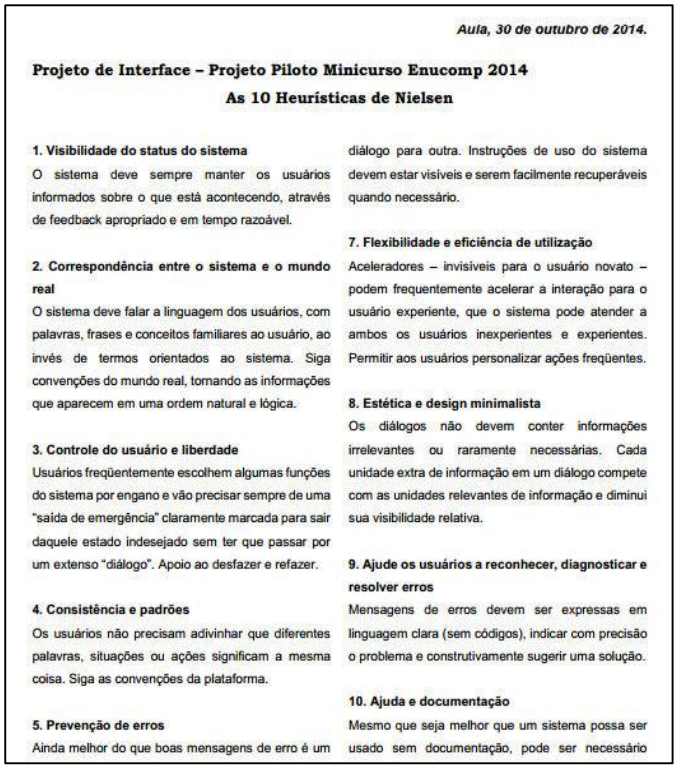
6. Experiências anteriores: relatos e resultado do Projeto Piloto

Este minicurso foi idealizado a partir da apresentação prévia de uma revisão sistemática sobre heurísticas de Nielsen para celulares, esta apresentação ocorreu no laboratório de estudos do usuário e de qualidade do uso de sistemas (LUQS) para pesquisadores da área de Interação Humano Computador (IHC), estavam presentes docentes e discentes de pós-graduação em Informática Aplicada e membros da Universidade de Fortaleza, Universidade Federal do Ceará, Universidade do Estado do Ceará e do Instituto Federal de Educação Ciência e Tecnologia do Pará, formando uma equipe multidisciplinar com Engenheiros da Computação, Tecnólogo em Processamento de Dados, Bacharéis em Ciência da Computação, Licenciados em Matemática e Designers. Após apresentação houve uma discussão e reflexão sobre a revisão sistemática que está em andamento e foi sugerido horizontes e perspectivas para esta pesquisa com base na experiência acadêmica e profissional de cada participante.

Uma das propostas foi a elaboração deste minicurso, contudo, realizou-se anteriormente um Projeto Piloto com a turma de Graduação em Ciência da Computação da Universidade de Fortaleza no mês de outubro para alunos matriculados na disciplina de Projeto de Interface.

Na data escolhida os alunos foram conduzidos para o LUQS em virtude de sua estrutura física ser mais apropriada para trabalhos em grupo do que a sala de aula. Os autores fizeram uma breve apresentação pessoal e indicaram suas áreas de pesquisa na pós-graduação. Houve um momento de aprendizagem sobre os seguintes conceitos: Qualidade de Software; 10 heurísticas de Nielsen e Inspeção de Usabilidade com apresentação de slides. Em seguida os participantes foram divididos aleatoriamente em grupos de trabalho e houve a exibição de um vídeo de inspeção de heurísticas do website do Instituto de Tecnologia de Massachusetts (MIT) disponível no YouTube, com pausas e comentários para um melhor entendimento dos participantes.

Imagem 1: recorte do material PDF utilizado como apoio.



Aula, 30 de outubro de 2014.

Projeto de Interface – Projeto Piloto Minicurso Enucomp 2014

As 10 Heurísticas de Nielsen

1. Visibilidade do status do sistema
O sistema deve sempre manter os usuários informados sobre o que está acontecendo, através de feedback apropriado e em tempo razoável.

2. Correspondência entre o sistema e o mundo real
O sistema deve falar a linguagem dos usuários, com palavras, frases e conceitos familiares ao usuário, ao invés de termos orientados ao sistema. Siga convenções do mundo real, tornando as informações que aparecem em uma ordem natural e lógica.

3. Controle do usuário e liberdade
Usuários freqüentemente escolhem algumas funções do sistema por engano e vão precisar sempre de uma "saída de emergência" claramente marcada para sair daquele estado indesejado sem ter que passar por um extenso "diálogo". Apoio ao desfazer e refazer.

4. Consistência e padrões
Os usuários não precisam adivinhar que diferentes palavras, situações ou ações significam a mesma coisa. Siga as convenções da plataforma.

5. Prevenção de erros
Ainda melhor do que boas mensagens de erro é um diálogo para outra. Instruções de uso do sistema devem estar visíveis e serem facilmente recuperáveis quando necessário.

7. Flexibilidade e eficiência de utilização
Aceleradores – invisíveis para o usuário novato – podem frequentemente acelerar a interação para o usuário experiente, que o sistema pode atender a ambos os usuários inexperientes e experientes. Permitir aos usuários personalizar ações freqüentes.

8. Estética e design minimalista
Os diálogos não devem conter informações irrelevantes ou raramente necessárias. Cada unidade extra de informação em um diálogo compete com as unidades relevantes de informação e diminui sua visibilidade relativa.

9. Ajude os usuários a reconhecer, diagnosticar e resolver erros
Mensagens de erros devem ser expressas em linguagem clara (sem códigos), indicar com precisão o problema e construtivamente sugerir uma solução.

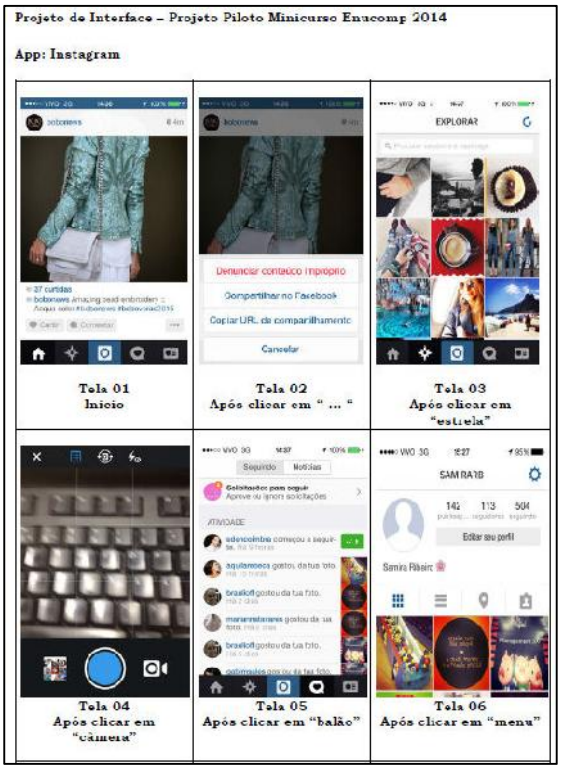
10. Ajuda e documentação
Mesmo que seja melhor que um sistema possa ser usado sem documentação, pode ser necessário

Imagem 2: recorte do material em slides utilizado como apoio.

Em seguida os 12 (doze) participantes realizaram inspeção de usabilidade baseada nas heurísticas de Nielsen das telas do aplicativo Instagram.

Foto 1: alunos realizando inspeção de usabilidade.

Imagem 3: recorte do material PDF com telas do Instagram.



Em seguida, pedimos para os alunos listarem entre as 10 heurísticas quais eles associaram como problema. Percebemos neste momento que as repostas divergiram e não houve nenhuma heurística como unânime na avaliação de cada grupo. Ficou claro que o teor de subjetividade causa estas respostas tão diferentes, contudo, durante a socialização no final da aula, tivemos relatos da turma de que a partir deste primeiro contato com o tema de usabilidade com foco em qualidade e partir de uma inspeção na prática, os participantes irão pensar de forma cautelosa quando estiverem projetando ou participando da concepção de alguma interface, seja ela web ou para aplicações móveis.

No final, foi aplicado um questionário para avaliar e coletar perfis dificuldades de aplicação e feedback sobre aplicação das heurísticas que também será aplicado ao final do minicurso como levantamento para um trabalho futuro que encontra-se em andamento.

7. Conclusões e trabalhos futuros

O minicurso proposto e previamente testado, oferece uma oportunidade de reflexão sobre a qualidade de aplicativos móveis a partir da técnica de IHC denominada inspeção de usabilidade com execução prática da mesma envolvendo os stakeholders do projeto. Estes conhecerão os principais conceitos relacionados às heurísticas e avaliação de usabilidade, bem como realizarão uma reflexão sobre o impacto de negligenciar aspectos de usabilidade sob o foco dos usuários. Este minicurso pode ser aplicado periodicamente nas empresas para a formação ou capacitação continuada de stakeholders interessados em melhorar suas atividades de desenvolvimento de aplicativos móveis e consequentemente a qualidade de uso desses aplicativos. Como trabalho futuro, iremos aplicar esta oficina numa empresa de desenvolvimento de software do estado do Ceará transformando-a em uma abordagem para empresas de pequeno e médio porte que pretendem investir em qualidade de seus aplicativos.

8. Agradecimentos

Este trabalho contou com o apoio do laboratório de estudos do usuário e de qualidade do uso de sistemas (LUQS) da Universidade de Fortaleza (Unifor), que cedeu espaço físico e equipamentos para execução de palestra e de um projeto piloto que deram origem ao minicurso submetido ao Enucomp 2014. Agradecemos também a co-autoria e orientação dos Professores Adriano Bessa e Elizabeth Furtado.

9. Referências

[1] Diego Henrique Dantas de Oliveira, Leonardo Cunha de Miranda, Erica Esteves Cunha de Miranda, Lyrene Fernandes da Silva. Prototipação de Interfaces de Aplicativos para Dispositivos Móveis: Estado da Arte e Desafios de IHC. Universidade Federal do Rio Grande do Norte (UFRN). IHC 2012.

[2] Antohy I. e Wasserman. Software Engineering Issues for Mobile Application Development. In Proc. FSE/SDP Workshop on Future of Software Engineering Research, ACM (2010), 397-400.

[3] Rocha, H. V., Baranauskas, M. C. C. Design e Avaliação de Interfaces Humano-Computador. NIED, UNICAMP, Campinas (2003).

[4] Prates, R. O., Barbosa, S. D. J. Avaliação de Interfaces de Usuário – Conceitos e Métodos. Juan Manuel Adán Coelho; Sandra C. P. Ferraz Fabbri. (Org.). Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação. Campinas: SBC, 2 (2003), 245-293.

[5] Matera, M., Costabile, M. F., Garzotto, F., Paolini, P. SUE Inspection: An Effective Method for Systematic Usability Evaluation of Hypermedia. In IEEE Transactions on Systems, Man and Cybernetics, Part A, 32 (2002), 93-103.

[6] L. Henrique, T. Lacerda, C. Gresse von Wangenheim, R. Araujo Barbalho, "Customizando Heurísticas de Usabilidade para Celulares".

[7] O. Machado Neto, M. daGraça Pimentel, "Heuristics for the Assessment of Interfaces of Mobile Devices".

[8] R. Inostroza, C. Rusu, S. Roncagliolo, V. Rusu, "Usability Heuristics for Touchscreen-based Mobile Devices: Update".

[9] Natasha M. Costa Valentim, Káthia Marçal de Oliveira, Tayana Conte. Definindo uma Abordagem para Inspeção de Usabilidade em Modelos de Projeto por meio de Experimentação. Universidade Federal do Amazonas. IHC 2012.

Honeypots e tecnologia mobile: descobrindo o invasor

Breno Fabrício Lira Melo Sousa[1]
Tiago Martins Ribeiro[1]
Raimundo Pereira da Cunha Neto[2]

Resumo: Desde a chegada do sistema operacional Android lançado pela Google, em 2008, tivemos um elevado ganho de processamento, memória e maior velocidade de conexão com a internet através da nova geração de celulares, os smartphones. Essa nova geração, trouxe consigo uma vastidão de benefícios e utilidades, como, a simples forma de lazer e passatempo – crianças, jovens e adultos passam várias horas do dia jogando, trocando mensagens com amigos, etc. – até mesmo a troca de e-mails, transações bancárias, que necessitam cada vez mais uma maior proteção das informações. Neste enfoque, usuários comuns cada vez mais estão adquirindo tais dispositivos. Entretanto, devem ter uma maior atenção e cautela em realizar atividades sigilosas, pois os mesmos, podem ser vítimas de invasores que porventura podem roubar informações, como: senhas bancárias, e-mails, etc. Todavia, existem formas de tentar prevenir tais furtos de informações, como: antivírus, VPN, honeypots, firewall, entre outros. Neste artigo abordaremos uma arquitetura em camadas para nosso honeypot móvel. Desde modo facilitará sua implementação e seu entendimento, que são as camadas: Gerenciamento, Log Component, Resposta e Coleta.

Palavras-chave: Honeypot. HoneypotLabsac. Honeypot mobile. Smartphone.

Abstract: *Since the arrival of the Android operating system launched by Google, in 2008, we had a high gain processing, memory and a faster internet connection through the new generation of cell phones, the smartphones. This new generation, brought with it a multitude of benefits and utility, as a simple form of entertainment and leisure - children, youth and adults spend several hours per day playing, exchanging messages with friends, etc. - even the exchange of e-mails, bank transactions, which require increasingly greater information protection. In this approach, ordinary users are increasingly purchasing such devices. However, should have greater attention and caution in performing sensitive activities, because the same can be victims of invaders that can possibly steal information, such as banking passwords, emails, etc. However, there are ways to try to prevent such thefts of information, such as: antivirus, VPN, honeypots, firewalls, others. In this paper we will talk about a layered architecture for our mobile honeypot. In this way will facilitate its implementation and understanding, which are the layers: Management, Log Component, Answer and Collection.*

Keywords: Honeypot. HoneypotLabsac. Honeypot mobile. Smartphone.

1 Introdução

A computação ao longo dos anos ganhou um enorme crescimento desde quando surgiram os primeiros computadores que tinham finalidades militares. As informações tornaram-se acessíveis para todos, a qualquer hora e lugar. Em seu processo de envoltório, os computadores foram ganhando cada vez mais melhorias, poder de processamento e armazenamento, e até mesmo de tamanho. Chegaram a um ponto de mudar o modo de vida da população mundial, dando uma maior comodidade e solucionando problemas que até então poderia ser quase que impossível de serem resolvidos sem a utilização dos mesmos.

[1] Ciência da Computação – Centro de Ensino Unificado de Teresina – Ceut – Teresina–PI {breno_fabricio23, tiagomartinz@hotmail.com}
[2] Mestre em Engenharia de Eletricidade – Universidade Federal do Maranhão – São Luís–MA {netocunhathe@gmail.com}

Todos os dias nos deparamos com máquinas cada vez menores e mais potentes. Comumente, podemos perceber que o crescente número de pessoas que possuem dispositivos com um grande poder de processamento e ar1mazenamento na palma da mão, proveniente da tecnologia mobile, cresce a cada dia.

Usuários cada vez mais estão trocando informações através da internet, que vai das mais simples transações, como a troca de e-mail, por exemplo, até transações confidenciais que merecem maior atenção e uma maior proteção de dados, como transações bancárias, por exemplo.

Para tanto, podemos utilizar o conceito de Sistema de Detecção de Intrusos (IDS), Firewall e honeypot para que possamos ter um maior conhecimento de qual técnica utilizar em um eventual ataque.

Com a utilização de um honeypot, como meio de defesa, nos deparamos com alguns questionamentos, que não podemos deixar de fora, como segue: honeypot é uma técnica de identificação de intrusos eficaz? Como identificar um intruso utilizando honeypot? Que tipos de serviços são os maiores alvos dos invasores?

Neste artigo, iremos propor uma nova arquitetura para o HoneypotLabsac, que se trata de honeypot mobile a nível de software. Essa nova arquitetura proposta será dividida em camadas, de modo que facilitará sua compreensão e implementação.

Iremos abordar na seção dois desse artigo honeypots voltados para dispositivos móveis. Seguindo, na seção três sua um breve histórico do surgimento dos honeypots. Na seção quatro, teremos uma abordagem sobre sua definição, características, classificações, honeynet, honeytokens e seus pontos fortes e fracos. Seguindo, iremos expor na seção cinco demonstraremos o sistema operacional android. Dando continuidade, na seção seis, abordaremos honeypots voltados a dispositivos móveis. Na seção sete teremos uma breve abordagem sobre o HoneypotLabsac. Na seção oito teremos uma nova arquitetura proposta para o HoneyLabsac. E por fim, teremos nossa conclusão e referências bibliográficas.

2 Honeypot voltado para dispositivos móveis

Em relação à honeypots para telefones móveis, ainda existem poucos trabalhos relacionados “devido aos recursos de hardware limitados dos dispositivos móveis e suas vulnerabilidades de software” [1].

Collin *et al*. (2011), citado por [2], criaram um honeypot para dispositivos móveis chamado de HoneyDroid. Tais autores, ao invés de trabalhar com o software em si, escolheram trabalhar diretamente com o hardware, pois assim eles poderiam ter uma maior visibilidade do honeypot.

Com a utilização do HoneyDroid, foram virtualizados memória flash, modem e WiFi, para que conseguissem controlar as interações ocorridas entre o sistema operacional android e o hardware do dispositivo. Deste modo, obtiveram um “monitoramento eficiente, gerando arquivos de log e armazenamento destes em um local não acessível ao sistema operacional Android” [2].

Como o HoneyDroid utilizou a virtualização para utilizar os serviços do sistema operacional Android, foram encontrados problemas de sobrecarga da CPU do dispositivo móvel. Devendo considerar que tal sobrecarga poderá ser percebida pelo atacante, isto é, tais autores, conforme [2], chegaram à conclusão que seria viável sua criação para aparelhos móveis, mas para tanto, teria que utilizar a virtualização para conseguirem um sistema completo.

Os autores [1] também relatam, que o HoneyDroid apresenta a desvantagem de não comportar-se como o legitimo sistema operacional Android. Tal desvantagem pode ser entendida como um malware, podendo ocasionar o encerramento do ataque, fazendo com que o atacante escape do honeypot.

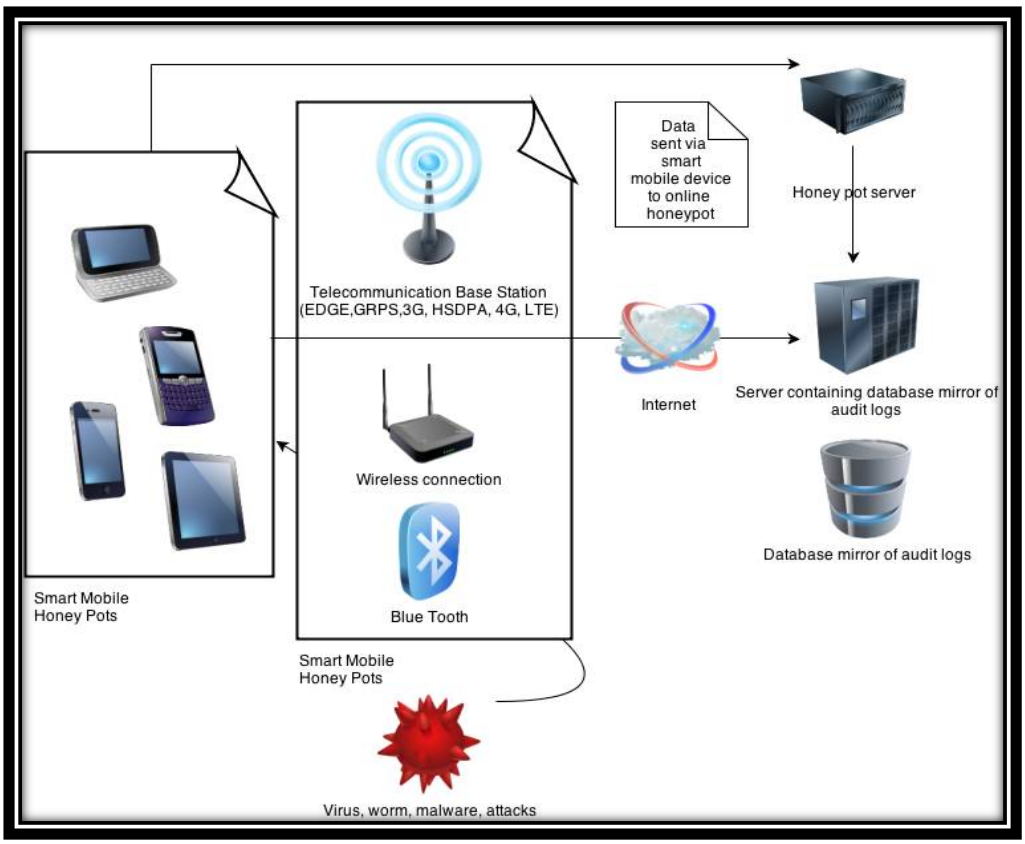
O’connor e Sangster (2010), mencionado por [2], desenvolveram um framework para HoneyClient (um tipo de honeypot) virtual para aparelhos móveis, com o objetivo de encontrar fragilidades ou “códigos maliciosos que afetem uma máquina ou um aplicativo cliente, por exemplo um navegador web (Browser)” [2].

De acordo com [1], vários problemas são enfrentados durante a criação de um honeypot para dispositivos móveis, como: a configuração do sistema, a parte monitoramento, contenção e visibilidade.

A configuração do sistema depende de como realmente desenvolver um sistema de honeypot para telefones celulares, e irá depender de qual sistema operacional de telefone móvel iremos desenvolver o honeypot. O monitoramento é a peça-chave, pois nosso honeypot só nos será útil se pudermos ter total visão do que ocorre na

rede, além de termos de saber o que o atacante está fazendo. A figura 1 representa um diagrama de rede de alto nível para honeypots móveis.

Figura 1: Diagrama de rede de alto nível para honeypots móveis



A contenção permite-nos ter o controle de que o honeypot não será utilizado como mecanismo para que o atacante faça ataques reais, assim comprometendo nosso sistema, e, finalmente a visibilidade que é muito importante, pois nosso honeypot terá que ser visível para nossos invasores. Teremos que ter informações atrativas, como por exemplo, "a publicação do número de telefone, endereço de e-mail, nome da conta de mensagens instantâneas e como em tantas maneiras possível" [1].

3 Histórico

Inicialmente, seu conceito foi apresentado por Cliff Stoll, com a publicação de seu livro, em 1990, "'The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage' (O ovo do cuco, rastreando um espião pelo labirinto da espionagem de computadores)" [3]. No ano posterior, Bill Cheswick publicou seu texto "'An Evening With Berferd, in which a Hacker is Lured, Endured, and Studied'" [3]. Nesse texto, Cheswick relata como conseguiu enganar um hacker com uma isca, e ao mesmo tempo estudá-lo.

Para que possamos ter um maior entendimento como surgiu essa poderosa técnica, teremos que retroagir no tempo e vermos quais fatores influenciaram em seu surgimento:

- Em 1997, o lançamento do Deception Toolkit, por Fred Cohen, onde foi considerado o primeiro honeypot, onde possuía seu código aberto e era grátis;
- No ano seguinte, surgiu o primeiro produto, o Sting, da antiga Cybercop, onde posteriormente foi adquirida pela NAI, no final do mesmo ano;
- Ainda no mesmo ano de 1998, Martin Rash, criou um honeypot para o governo norte americano;

De acordo com Oliveira Junior, Deco e Antonio (2004), em 1999, surgiu o Honeynet Project, criado por Lance Spitzner em uma entidade formada por cerca de 50 especialistas de segurança.

4 Honeypot

Honeypots são ferramentas utilizadas para a monitoração de ataques, coletando informações importantes sobre tendências e permitindo aprimorar a metodologia utilizada para a segurança de uma empresa [4].

Temos, ainda, que salientar que, diferente de um Firewall ou IDS (Sistema de Detecção de Intruso), um honeypot não irá resolver apenas um ou outro problema específico na falha de segurança, mas sim interpretar a rede como um todo, ajudando assim a outros mecanismos de defesa a identificar onde e quais falhas um determinado sistema possui.

Para [5] honeypot trata-se de uma ferramenta que não se engloba com uma ferramenta de segurança ou de prevenção de ataques, mas sim de estudo, no qual através desta podemos conseguir informações detalhadas sobre um eventual atacante.

Conforme [6], destaca que honeypots são armadilhas para invasores. Configura-se um honeypot com falhas de segurança afim de ser uma isca para o invasor. Após ser comprometido, este irá coletar informações sobre o invasor.

4.1 Características

Em relação às suas características, podemos nos deparar com os seguistes honeypots: honeypots de produção e de pesquisa [4]. Os "Honeypots de produção são utilizados para distrair a atividade maliciosa de máquinas com maior valor na rede ou como um mecanismo de alerta" [4]. Ainda temos que ter em mente que, os honeypots de produção poderão dar auxilio para mecanismos de segurança, como por exemplo, IDS e Firewall [7]. Tais honeypots são mais fáceis de serem implantados, pois apresentam um menor número de funções, trazendo consigo uma menor taxa de risco para o sistema. Entretanto, a obtenção de informações que o honeypot de produção irá conseguir coletar será menor, em comparação com o honeypot de pesquisa.

Diferenciando-se dos honeypots de produção, os honeypots de pesquisa "são utilizados para a monitoração de um ataque com o objetivo de capturar o maior número de dados possíveis para posterior análise" [4].

Para tanto, o intuito para o desenvolvimento de honeypots de pesquisa, é a obtenção diretamente sobre os atacantes, pois não irá concentra-se em apenas uma única organização [7].

Contudo, seu principal objetivo é obter a maior capacidade de informações possíveis de um ataque, não deixando de capturar "quem são os atacantes, como eles estão organizados, de onde ocorrem os ataques, que ferramentas são utilizadas e como são obtidas essas ferramenta" [7].

4.2 Classificação

Honeypots podem ter uma grande utilidade para as empresas, uma vez que seu foco principal é enganar o invasor, dando-lhe um sistema ou qualquer outro serviço previamente configurado para ser invadido, para que possam ser estudados quais técnicas e mecanismos o invasor utilizou durante a invasão.

Ressalta-se que, durante o processo de configuração de um honeypot, temos que configurá-lo de maneira que o invasor não perceba que o mesmo esteja sendo monitorado, pois caso haja essa falha, o invasor irá fugir de nosso honeypot, tornando-o desnecessário.

Esse estudo só nos é permitido através de logs que são gerados constantemente em um honeypot, em que, cada log contém, no mínimo, as referidas informações: "data e hora do ataque, IP de origem (atacante) e IP de destino (atacado) e tipo de ataque" [2].

Uma vez gerados os logs, estes são armazenados, para que o(s) administrador(es) de rede possam analisá-los futuramente, podendo ou não modificar suas configurações para uma maior proteção, pois caso haja uma falha real, o honeypot pode servir de ponte para que o invasor consiga se infiltrar no sistema real.

Encontramos honeypots em dois níveis de classificação: os honeypots de baixa interação e os honeypots de alta interação, em que, os de baixa interação irão prover falsos serviços, e ocorrerá interação com o invasor, dando-lhe falsas informações [2].

Diferenciando-se dos de baixa interação, os honeypots de alta interação, irão disponibilizar um ambiente real para o invasor, onde o mesmo poderá interagir, tanto com o sistema operacional em si, ou com aplicações ou serviços da empresa. Nesse cenário, teremos que ter uma maior cautela, pois caso aconteça alguma falha de segurança, o sistema poderá ficar comprometido [2]. Nesses níveis de classificação, o autor [2] destaca que alguns autores ainda consideram uma classificação intermediária de honeypots, situando-se entre os de baixa e alta interatividade.

Honeypots de classificação intermediária irão prover serviços mais ricos em informações do que a camada de baixa interatividade, afim de proporcionar uma maior e melhor ilusão de um sistema fácil de ser invadido. Neste tipo de classificação, administradores de rede devem ter uma atenção redobrada, pois, serviços real do sistema podem ser utilizados nesse nível.

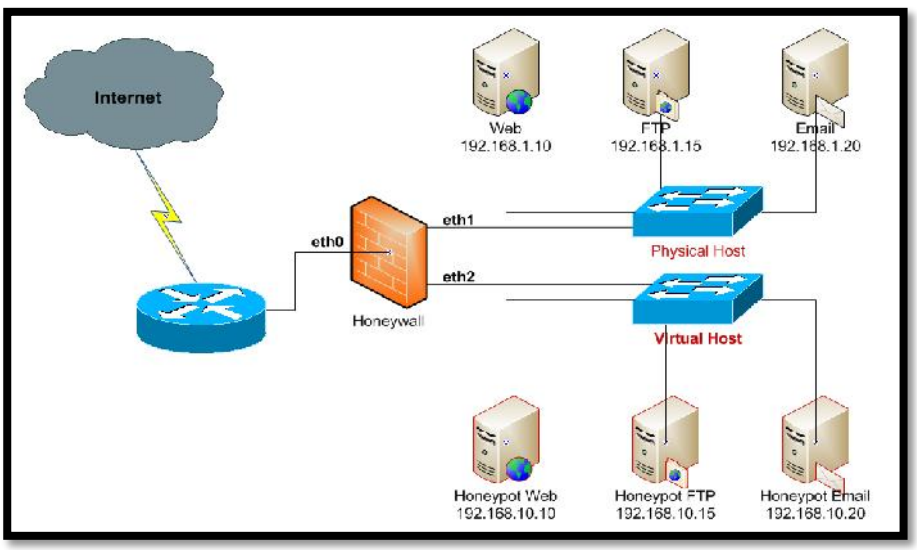
Tabela 1: Níveis de classificação

Classificação	Captura de dados	Risco de invasão
Baixa interatividade	Limitada	Baixo
Média interatividade	Moderado	Médio
Alta interatividade	Extensa	Alto

4.3 Honeynet

Conforme [4], uma honeynet é um aglomerado de honeypots, tornando-se assim uma rede virtual de computadores, com o objetivo de serem comprometidos. Tal comprometimento da rede, servirá como mecanismo de estudo para observar o comportamento dos invasores, possibilitando-se, assim, uma análise mais aprofundada das ferramentas utilizadas, o objetivo do intruso e quais vulnerabilidades foram, de fato, exploradas. Na figura 2, teremos um exemplo de uma honeynet, por meio da qual poderemos ter um maior entendimento de como ela se constitui.

Figura 2: Exemplo de honeynet



Assim como nos honeypots, todo o tráfego de rede de uma honeynet é capturado e armazenado em arquivos, com o propósito de serem posteriormente estudados, pois assim como num honeypot, ocorrendo uma real falha de segurança, uma honeynet poderá ser utilizada como porta de entrada para um sistema real, como podemos observar:

> Todo o tráfego de uma Honeynet gera registros para serem estudados, pois qualquer conexão efetuada é monitorada, facilitando a detecção de riscos e não ocasionando nenhum impacto para a rede corporativa em uso. Com as Honeynets podemos aprimorar a capacidade de detecção, reação, recuperação e análise dos sistemas. Todas as técnicas utilizadas são submetidas a constantes análises após cada ataque realizado para que possam ser aperfeiçoadas. [4]

4.4 Honeytokens

Honeytokens são dados e/ou informações disponibilizadas de forma atrativa, a fim de chamar a atenção de um invasor. Podendo ser de uma simples falha na segurança de um determinado sistema da empresa, ou até mesmo 'informações' disponibilizada propositalmente pelo administrador de rede.

Os autores [8] destacam que um honeytoken não é honeypot, pois eles podem ser meios digitais que atraem a atenção de invasores. Alguns exemplos básicos de honeytokens que podemos nos deparar em uma honeynet, são: número de cartão de créditos, arquivos de planilhas eletrônicas, arquivos de apresentação, falsas falhas de segurança (senhas fracas de e-mails, por exemplo).

4.5 Pontos fortes e fracos

O autor [2] expõe que alguns autores destacam algumas das vantagens da utilização de honeypots, como: pequenos conjuntos de dados, novas ferramentas e táticas, captura de informações, facilidade de uso e encriptação IPv6.

Para pequenos conjuntos de dados (porém de grande valor), qualquer informação obtida em nosso honeypot nos trará grande utilidade, pois, qualquer informação gerada por ele, representa que algum intruso tentou realizar alguma ação não autorizada.

Em novas ferramentas e táticas, os administradores de redes, poderão ficar a par de novas formas de invasão, já que o honeypot irá gerar logs que serão estudados posteriormente.

Em relação à facilidade de uso utilizando os honeypots de baixa interatividade, devido às suas características, não precisaremos de uma maior complexidade em sua implantação.

A utilização de protocolo IPv6 não irá impedir com que honeypots capture informações sobre o tráfego de rede em um determinado ambiente, pois este captura qualquer informação que estiver sendo enviado ou recebido independentemente do que seja trafegado por um honeypot.

Como desvantagem, uma vez que nosso ambiente de honeypot seja comprometido, caso não tenha sido configurado de maneira correta, o intruso poderá utiliza-lo como porta de entrada comprometendo o sistema real da empresa.

> Uma das desvantagens é a visão limitada ao tráfego, pois a ferramenta pode detectar somente os ataques direcionados a ele, sendo assim ignorando as atividades relacionadas a outros sistemas. Portanto, ele não consegue detectar roubo de arquivos confidenciais feitos de usuários da própria rede e não consegue detectar ataques contra seu próprio servidor de serviços da rede [...]. (BORGES; BENTO, 2006, p. 31)

5 Sistemas operacional android

Sistema operacional Android é completo para a arquitetura de aparelhos móveis, pois foi desenvolvida essencialmente para smartphones, em que sua plataforma constitui-se dos seguintes componentes: sistema operacional, middleware, aplicativos e interface com o usuário [2].

No ano de 2008, o sistema operacional android foi lançado no mercado dos celulares, desenvolvido pela empresa Google. Em seguida, a mesma criou vínculos com as mais variadas empresas no ramo de telefone móvel, dando assim continuidade em seu projeto. Esse vínculo foi rotulado de Open Handset Alliance (OHA), contando atualmente com 84 empresas [2].

A base de fundamentação do Android, como menciona [2], é baseada e fundamentada no kernel do sistema operacional Linux. Porém, algumas alterações tiveram de serem feitas, desde o seu surgimento, para que apresentasse características de um telefone móvel, tais como [2]:

- Binder – que é utilizado pela comunicação de processos. Garantirá que nenhum processo tenha acesso ao espaço de memória dos demais processos;

- Ashmem – caracterizado como uma nova maneira de se ter memória compartilhada entre dois processos, possibilitando que os mesmos se comuniquem por essa região de memória compartilhada;
- Wakelocks – ocorre a detecção se o aparelho está sendo usado ou não; caso não esteja, converterá para o modo de economia de energia;
- Oomhandling – ficará responsável por controlara utilização de memória do sistema, e encerramento dos processos, caso não aja memória disponível para sua execução.

6 Ameaças aos dispositivos móveis

Podemos ter em mente que ameaça aos dispositivos móveis é algo recente, surgida com a chegada dos smartphones, mas na realidade não é essa a verdade.

Os perigos enfrentados pelos usuários de aparelhos celulares não é algo recente. O primeiro vírus digital criado para telefones móveis surgiu no ano de 2004, o "Cabir", que era espalhado via Bluetooth, e tinha como foco exclusivo atacar o sistema operacional Symbiam [2].

Devido ao fato de uma não padronização universal de tais ameaças, Dunham, K.,et al (2009), citados por [2], fazem uso de termologias próprias para a definição de segurança móvel, como segue:

- Ad/Spyware – programas indesejáveis, que podem executar várias ações sem a autorização do usuário;
- Bluebug – sua função é explorar vulnerabilidades no Bluetooth, para efetuar chamadas telefônicas com valor mais caro.
- BlueChop– constitui-se na negação de serviço de uma rede Piconet[1] ;
- Denial-of-Service (DoS) – caracteriza-se num ataque com o objetivo de prejudicar e/ou negar o uso de um aparelho móvel, serviço ou rede;
- Exploit – poderá se caracterizar tanto como um software ou ações que visam utilizar aberturas de um sistema para executar ações não autorizadas;
- Hacking default – sua função é invadir dispositivos ou softwares que possuem senha, definições ou configuração padrão, como segurança.
- MalwareMóvel – é um software que executa ações maliciosas em dispositivos móveis.
- Snarf – é o furto não autorizado de dados.
- Trojan – Software que proporciona atividade maliciosa e, se camufla como algo que não é.
- Vírus – Software que irá proporcionar atividades maliciosas que tem como função propagar-se entre dispositivos, danificando seu funcionamento.
- Worm – Diferente dos Vírus, duplicam-se com objetivos de espalharem-se pela rede.
- Força Bruta – Consiste em exaustivas combinações para tentar encontrar a chave de acesso (senhas, por exemplo).

Como podemos perceber, foram expostas algumas das mais variadas ameaças com os quais podemos nos deparar comumente atualmente. Conforme [2], embasado por Enck, Ongtang e McDaniel (2009), as ameaças relatadas, tiveram sua base nos computadores pessoais, os quais tiveram que sofrer algumas alterações para que pusessem ser utilizadas nos aparelhos celulares.

7 HoneypotLabsac

HoneypotLabsac é o primeiro honeypot móvel a nível de software. Está classificado como um honeypot de baixa interatividade, em que, faz uso de emulação dos serviços de Telnet e http, em que, tal registro de conexões serão gravados em arquivos de logs e "envios e recebimentos de SMS transmitidos pela conexão emulado do Telnet" [2].

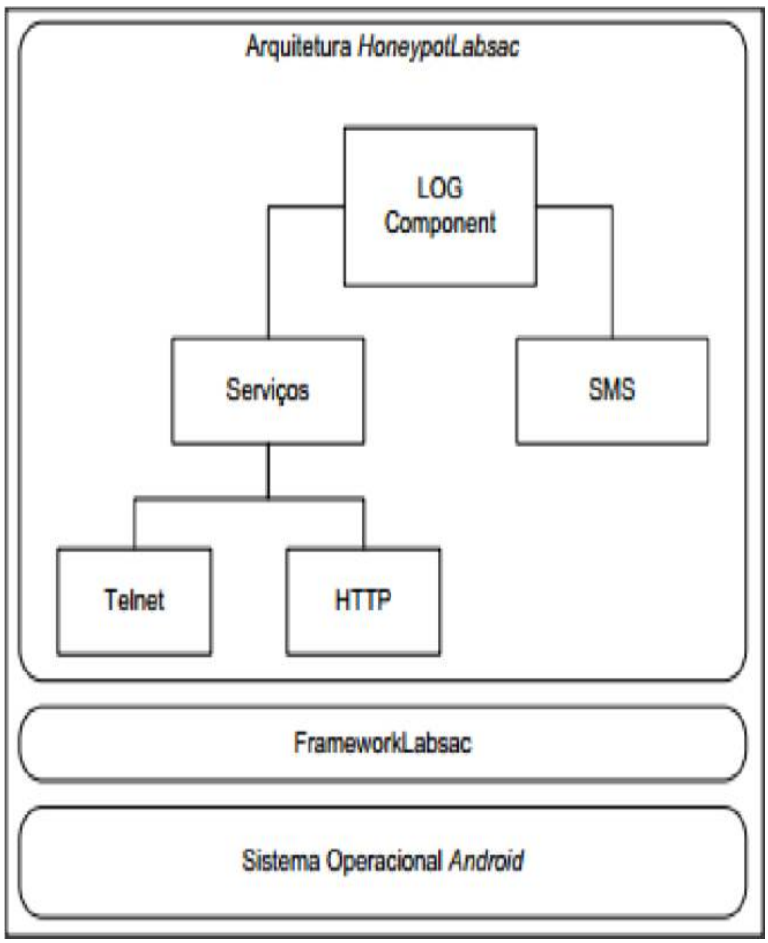
- No log de registro de atividades que foram executados, são apresentadas as informações de [2]:

[1] Uma rede Piconet trata-se de uma rede Bluetooth composta de no máximo 8 dispositivos. Um dispositivo mestre e sete escravos.

- Data e hora da conexão feita ao dispositivo;
- Endereço de IP e porta de origem do atacante;
- Endereço de IP e porta do alvo;
- E comandos digitados.

Para que possamos ter uma melhor noção de sua arquitetura, visualizemos a figura 3.

Figura 3: Arquitetura HoneypotLabsac



Estaremos utilizando por base uma nova arquitetura para o HoneypotLabsac, que foi proposto por [9], em que, foi acrescentado o serviço de emulação de *Bluetooth*. Esta arquitetura apresenta quatro camadas: gerenciamento, log component, resposta e coleta. Com base nesta nova arquitetura, estaremos propondo a inclusão de um novo serviço, serviço de troca de mensagens do *WhatsApp*.

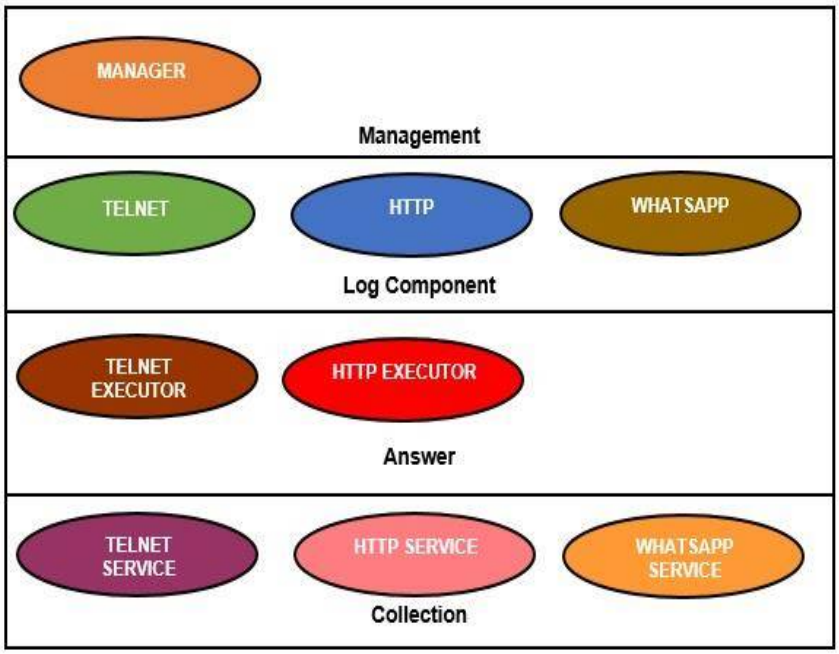
Na camada de gerenciamento, será realizada a tarefa de capturar os logs criados pelos serviços da cama de Log Component, e transferi-los para um ambiente seguro (fora do dispositivo).

A camada Log Component, tem como funcionalidade, criar e salvar os logs dos serviços que estão sendo utilizados, em que cada tipo de serviço terá seu log específico.

A camada de resposta, tem como finalidade de prover uma resposta para o atacante quando este solicitar algum dos serviços emulados, em que cada serviço dará uma resposta correspondente ao seu tipo de serviço.

E por fim, a camada de coleta, que disponibilizará os serviços que estarão sendo emulados pelo nosso honeypot, que emulará os serviços de Telnet, Http, e WhatsApp.

Figura 4: Arquitetura em camadas



8 Conclusão

A nova proposta de arquitetura para o HoneypotLabsac tem como principal a característica de simplicidade organizacional que será dividida em camadas e, bem como a facilidade de implementação, uma vez que sua nova estrutura não irá alterar o funcionamento dos demais serviços.

Contudo, percebemos que honeypots podem ser de grande utilidade na defesa contra ameaças externas de uma empresa ou telefones móveis. Sua limitação de captura de dados irá depender da sua configuração.

Como podemos perceber, telefones celulares tornaram-se alvo de invasores a partir de 2004, com o advento dos denominados smartphones. Com o passar dos anos, foram ganhando cada vez mais poder de processamento, funcionalidades que antes não possuíam.

Tornaram-se cada vez mais essencial no cotidiano da população mundial, necessitando cada vez mais proteção das informações dos usuários, pois devido às grandes funcionalidades que tais aparelhos são capazes de ter, caso haja um roubo de suas informações, o mesmo poderá sofrer vários problemas, uma vez que em seu aparelho telefônico, o usuário poderá ter dados e senhas sigilosas salvas.

Referências

[1] AHMED, H. M; HASSAN, N. F; FAHAD, A. A. *A Suvery on Smartphone Honeypot*. ISSN 2277-3061.2013.

[2] OLIVEIRA, V. B. *HoneypotLabsac: um framework de honeypot virtual para o android*. 2012. Dissertação (Mestrado em Engenharia de Eletricidade) - Universidade Federal do Maranhão, São Luís, 2012. Disponível em: <http://www.bcc.unifalmg.edu.br/bibliotecabcc/files/Discenes/Monografias/Monografia_LucasTardioli.pdf>. Acesso em: 16 nov. 2013.

[3] SIMÕES, S.; SILVA, F. *Estudo de ferramentas para honeypots instaláveis em máquinas virtuais perfazendo uma honeynet virtual*. Curitiba, 2009. Disponível em: <http://www.ppgia.pucpr.br/~jamhour/RSS/TCCRSS08A/Sidney%20Simoes%20e%20Silva%20Filho%20%20Artigo. pdf >. Acesso em: 16 nov. 2013.

[4] OLIVEIRA JUNIOR, C. M.; DECO, F. S.; ANTONIO, S. S. *Honeypots: enganando e conhecendo o inimigo*. 2004. Monografia (Graduação em Bacharelado em Informática) - Universidade do Grande Rio, Duque de Caxias. Disponível em: <http://www.apostilando.com/download.php?cod=3164>. Acesso em: 16 nov. 2013.

[5] BRAGANÇA, E. C. M; SILVA, H.C.C; SOUZA, J. R. *O uso de honeypots para estudo e combate de spam*. São Paulo, 2010. Disponível em: < http://www.aocubo.tecnologia.ws/wp-content/uploads/2011/07/O-uso-de-Honeypots-para-Estudo-e-Combate-de-SPAM.pdf>. Acesso em: 21 set. 2014.

[6] MATOS, F. M. A. *Proposta de um checklist para verificação da segurança física de uma empresa baseada na norma abnt nbr iso/iec 27002:2005*. Fortaleza, 2010. Disponível em: <http://www.flf.edu.br/revista-flf/monografias-computacao/monografia_marcelo_matos.pdf>. Acesso em: 20 set. 2014.

[7] SILVEIRA, L. T. *Detecção de intrusão através de configuração de honeypot de baixa interatividade*. 2011. TCC (Graduação em Bacharelado em Ciência da Computação) - Universidade Federal de Alfenas, Alfenas. Disponível em: <http://www.bcc.unifalmg.edu.br/bibliotecabcc/files/Discentes/Monografias/Monografia_LucasTardioli.pdf>. Acesso em: 16 jul. 2014.

[8] POUGET, F.; DACIER, M.; DEBAR, H. *Honeypot, honeynet, honeytoken: terminological issues*. Sophia Antipolis, 2003. Dísponível em: < http://www.eurecom.fr/fr/publication/1275/download/ce-pougfa-030914b.pdf>. Acesso em: 02 set. 2014.

[9] SILVA, K. K. S. *Método de identificação de intrusos baseado em honeypot para dispositivos móveis*. 2014. Monografia (Tecnólogo em Análise e Desenvolvimento de Sistemas) – Faculdade de Ciências e Tecnologia do Maranhão, Caxias.

Educational game platform as a tool for public schools in Picos - Piaui.

Bruno Pereira da Fonseca[1]
Francisco Imperes[1]

Abstract: In the past years schools have been trying to introduce new technologies to improve the learning process. Nowadays technology has become more and more available, and schools in Brazil have already started to acquire computer laboratories, although the absence of appropriate educational software has made these machines fail to their potential on public education. On the basis of this problematic, the article proposes the development of an educational tool for a game to be played by students and additional web-based application to be used by teachers, aiming to become an interactive and competitive platform that differentiates itself allowing its use on multiple school subjects, in which teachers control the content and on-game questions and work closely along with the classroom content. For the development of the proposed applications, multiple technologies are put to work together, including: HTML5, Ruby on Rails Framework, HTML5 game engine Construct 2, and the database manager MySQL. During the content of this article the aspects related to the software development, technologies integrations and achieved results will be discussed.

Keywords: HTML5, educational games, Construct 2.

1 Introduction

Video games can be used for much more than entertainment, and they have already been used in fields such as advertising, education and health. Throughout this research, the outcome is the development of a game that promotes the access to unconventional education activity. The game will be available, free of charges, for all the public schools in Picos, Piaui (Brazil), in an effort to become a helpful tool in the improvement of the education levels of the city.

The usage of games in education is not a new concept, but the usage of computers and digital games surely brings a new world of possibilities in the field of education. According to Barbosa & Muralloli (2013) [1],*"computers are becoming more and more presents in the daily life of our society. Its cultural presence grows every day and, with its arrival into schools, we need to think about what to expect from this technology to be used in the learning process"*[2]. Whenever an educational game is well designed it may be used as an additional aid to teaching practices, enhancing learning rates. The benefits and potentials of this new kind of media are wide, and being studied by researchers [2].

[1] Curso de Sistemas de Informação, UFPI, Campus Senador Helvídio de Barros – Picos (PI) - Brasil

[2] Translated by us. Original quote: "*os computadores estão cada vez mais presentes na vida cotidiana da nossa sociedade. Sua presença cultural aumenta a cada dia e, com a chegada às escolas, é necessário refletir sobre o que se espera desta tecnologia como recurso pedagógico para ser utilizado no processo de ensino-aprendizagem*" – Barbosa & Muralolli (2013).

Figure 1: Developing Index of Basic Education in Picos

Picos public educational level has raised humbly over the past years but is still overall low. According to [3] IDEB (Developing Index of Basic Education) of the Brazilian government, Picos' public primary education has increased its average from 2.7 points in 2005 to 3.7 in 2011 (Figure 1). Yet, it is still under the national average, ranking at 4.7 in 2011, which concludes that there is still a lot to be done in terms of quality of education. It's believed that when technology is properly applied in schools, it can make a difference and contribute to education. Thereby this educational tool aims to create an environment that stimulates children to learn, increasing their interest for the materials learned in the classroom and introducing a healthy competition amongst the children–all the while, allowing teachers to elaborate the content put into the game.

This paper is divided in three more sections: section number 2 describes and justifies the technologies and tools involved on each of the applications; section number 3 describes the way both softwares are integrated and the aspects related to the cross domain communication; section 4 describes early results and displays what has already been achieved in the research.

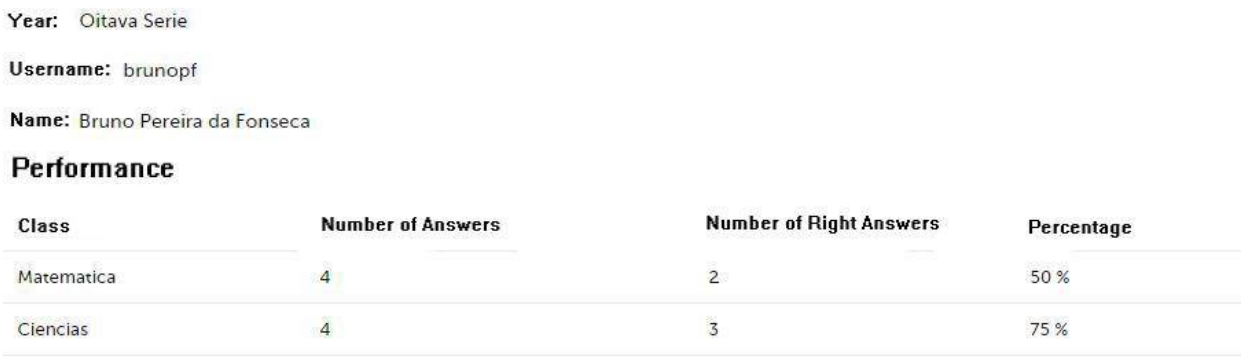
2 Technologies & Tools

The product of this project is two software products (one to be used by teachers and another to be used by students) because the tools used to create games is focused only on game development, which doesn't enable developers to construct a functional website (therefore the usage of a second technology). Both software's are web-based and communicate directly with each other, establishing a co-dependency and exchange multiple messages. This also requires one computer in the school labs to be chosen as host of the applications, which can be accessed in other computers through local network[3]. The technologies involved with each one of the applications and how they communicated are listed below:

2.1 Base Application for Teachers

The web application developed to be used by the teacher includes multiple functions related to the game aspects. In this application teachers can list, edit and create new school classes, school subjects and more importantly, each of the questions used in the game. Teachers can also list and visualize the students who are registered; their answers to the questions made in-game and also check a table with the performance of the students in each of the subject's questions (Figure 2).

[3] Usage of local network instead of the internet has been decided over connection limitations in some computer laboratories, though internet usage is available.



Year: Oitava Serie

Username: brunopf

Name: Bruno Pereira da Fonseca

Performance

Class	Number of Answers	Number of Right Answers	Percentage
Matematica	4	2	50 %
Ciencias	4	3	75 %

Figure 2: Performance of a student in the teachers' application.

Developed in Ruby in Rails, a web framework developed in 2003 by David Heinemeier Hanson [4], the framework already has tens of thousands applications. According to Tate & Hibbs [5], "*The most common problem in today's typical development project involves building a web-based user interface to manage a relational database. For that class of problems, Rails is much more productive than any other web development framework"*. Based on a MVC model (*Model-View-Controller*) and favoring convention over configuration, the framework has been chosen its wide usability, easiness on creating simple applications quickly, and also the support for AJAX & JSON (used to communicate the platform with the game).

Ruby on Rails allows the user to choose the Data Base Management System to be used in the application. For this application, MySQL was chosen since it is one of the most commons data base management systems available. In addition, MySQL offers a best-of-all worlds scenario: It runs on many platforms, enjoys a low TCO, and is stable [6].

2.2 Game for students

Developed in HTML5, this technology was selected because of performance aspects. Since some computer laboratories may have old computers with humble configurations, it's expected a technology that doesn't require too much processing. [7] "*HTML5 is the next generation of HTML, superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 provides new features that are necessary for modern web applications. It also standardizes many features of the web platform that web developers have been using for years, but have never been vetted or documented by a standards committee"*. One of the new elements introduced to HTML5 was the Canvas element, which can be used to display media content natively in browsers, without the need of third-party plugins. This lack of plugin has elevated the performance of online gaming considerably.

Creating games in HTML5 is not an easy task, but new game engines based in HTML5 already have demonstrated big potential to this technology. Hence, the game is developed in the main game engine available in the market: Construct 2, developed by Scirra[4]. There are many advantages in the usage of Construct 2 for game developers, such as [8] powerful event system; flexible behaviors, instant preview; visual effects; multiplatform export (including all mobile platforms, Facebook[5], Chrome Store[6] and even game consoles like WiiU[7]).

The structure of the technologies in both sides of the applications is represented by Figure 3.

[4] Scirra, more information available at https://www.scirra.com/
[5] Facebook, available at www.facebook.com
[6] Chrome Store, available at https://chrome.google.com/webstore
[7] WiiU, developed by Nintendo. More information at http://www.nintendo.com/wiiu

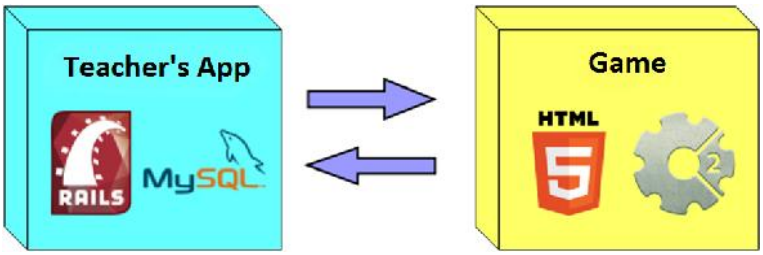


Figure 3: Structure of the Technologies used.

3 Software Integration: Cross-Domain Communication, AJAX and JSON

Since the two applications are web-based and Construct 2's restrictions of web server, both applications are running in different ports. Therefore, the communication between the two applications is done through cross-domain communication (Figure 4), which means that the domain of the web-app needs to send messages to the domain of the game, and vice versa. As a part of the web application security model, requests between apps are initially only allowed to same-origin policy. This is an important security measure for most websites, but since these applications are running locally and they are co-dependent, different configurations had to be done in the Ruby on Rails application to allow requests from the game. The information exchanged between the apps includes user login and registering, questions, answers and highscores.

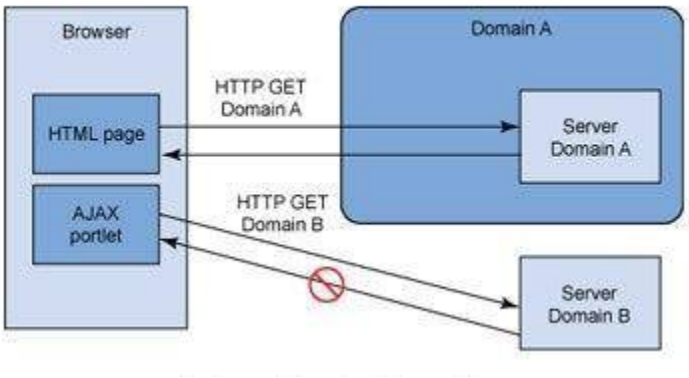


Figure 4: Example of Cross-domain requests. Source: CodeProject.com [8]

The following lines were needed to be listed in the Controllers of the Ruby on Rails application to allow cross domain requests, define request methods and content-type:

Code 1: Controllers of the Ruby on Rails application to allow cross domain requests

```
1. before_filter :set_access_control_headers
2. def set_access_control_headers
```

[8] Available at http://www.codeproject.com/Articles/158755/ASP-NET-JSON-Proxy-JSON-Web-Service-and-JavaScript . Accessed in September, 2014.

```
3.      headers['Access-Control-Allow-Origin'] = '*'
4.      headers['Access-Control-Request-Method'] = 'GET, OPTIONS, HEAD'
5.      headers['Access-Control-Allow-Headers']  =  'x-requested-with,Content-
Type, Authorization'
6.      headers['Content-type'] = 'application/json'
7. end
```

All communication between the platforms is made in runtime, which means that while the student is playing the game, a bunch of information is being exchanged in the background in an asynchronous way. Figure 5 demonstrates how asynchronous communication works in comparison with the synchronous communication: synchronous communication is linear where the sender sends the data to the receiver; but asynchronous communication is done in "*steps"* and can be done at any point. Asynchronous requests are made from the game to the framework, so the game can request needed information and, since all the game information is stored in the framework, the game needs to request data storage. These requests in runtime are only possible thanks to AJAX, a group of techniques that was created by the mix of JavaScript and XML (although XML usage is not required), which allows web applications to send data to, and retrieve data from a web server asynchronously without interfering with the display and behavior of the existing page. [9] AJAX incorporates:

- Standards-based presentation using XHTML
- Dynamic display and integration using the Document Object Mode
- Data interchange and manipulating XML and XSLT
- Asynchronous data retrieval using XMLHttpRequest
- and JavaScript binding everything together.

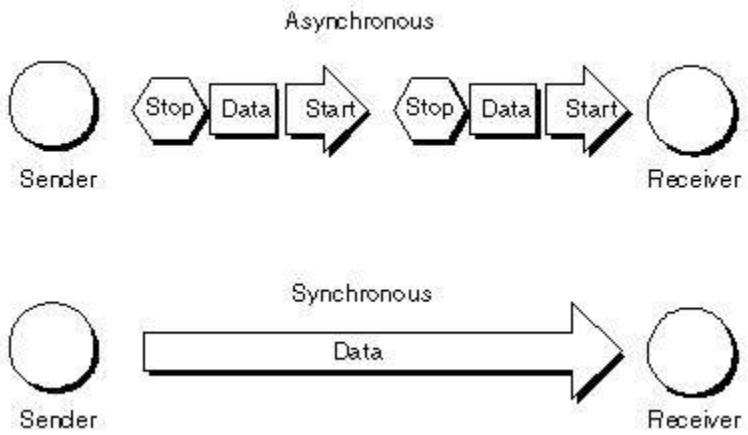


Figure 5: Asynchronous communication schema between server/client. Source: VendorSeek.com[9]

AJAX requests are handled in different way in Ruby on Rails and Construct 2. While in Ruby on Rails, AJAX requests are handled as functions; Construct 2 deals with AJAX as a Plugin. Construct's event system recognizes events such as AJAX request, *on success, on failure*. The major problem with Construct's AJAX plugin is that there is no personification of the AJAX headers – requests are handled with the same headers for every occasion.

While AJAX provides the possibility to exchange messages between the two cross domain applications, Ruby on Rails and Construct 2 are totally different technologies that don't understand each other. In fact, Construct 2 only has an event-based system and no lines of either code or variable types. JSON (*Java Script Object Notation)*

[9] Available at http://www.vendorseek.com/ajax-a-gift.asp . Accessed in September, 2014.

was used to enable the data that comes from the server to the game to be understandable through both parts. Such format is [10] completely language independent and it's available for most programming languages.

Construct 2 also differentiates itself in usage of JSON. The software deals with JSON's requests as Dictionaries, plus it also has its own structure. The JSON requests need to be adapted to follow the format of Construct's dictionary, which includes identifying that the JSON belongs to a Construct 2 Array, separating it from the data, for example:

Code 2: JSON's representation format in Construct 2 dictionaries

```
1. {"c2array":true,"size":[2,2,1],"data":[[[1], ["Oitava Serie"]], [[2],
2. ["Setima Serie"]]]}
```

4 Early Results & Demos

The development process of both software products started in August, 2014 and its due on December of this year. Although both softwares remain unfinished, and efforts are being made to improve GUI & human-computer interaction aspects, multiple functions are already functional.

4.1 Base Application for Teachers

The application developed in Ruby on Rails (Figure 6) is already running; for example, the manipulation of questions, students, in-game answers, questions, classes and subjects.

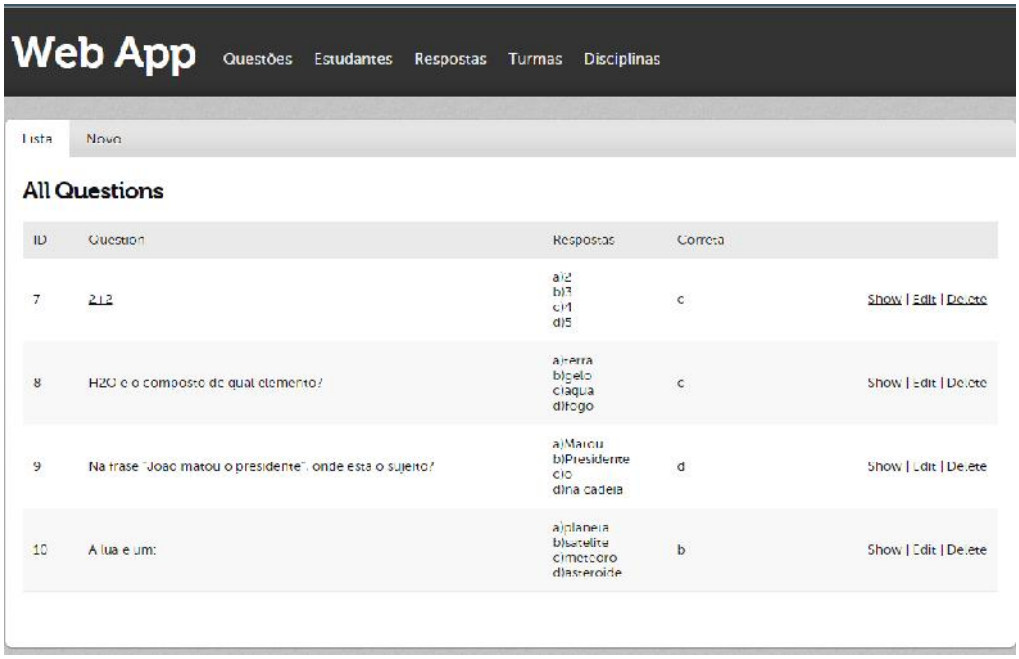


Figure 6: Overview of the demo for the website. The modules listed in the top of image are all the modules presented to an authenticated teacher.

Each one of the modules listed in Figure 6 present a list of the data stored; they also provide functions such as show, edit, delete and new. Questions are defined in the format of multiple alternatives with four different options (a,b,c,d). Figure 7 represents the form where questions are added:

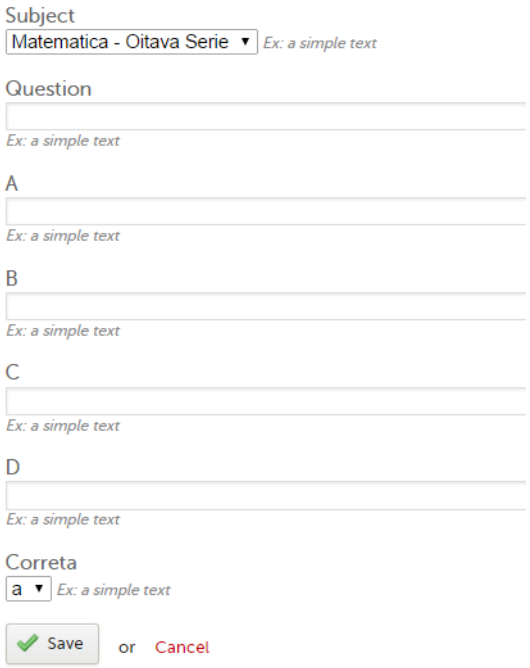


Figure 7: Overview of the form to add questions to the game.

Most communication aspects of how the Rails application provides and stores all the data used in the cross-domain have been implemented:

- Providing a list of years available for students to register in the current year;
- Giving the game questions corresponding with the year of the student playing it;
- Storing each answer given by student, regardless of whether it was correct or wrong;
- Storing each user's score and devise a "high-score table" sorting them by order, in an effort to make the student compete for the highest position.

4.2 Game for Students

Even though game developed to the students hasn't been tested with the targeted audience yet, the early demo already implements some major of the existing functions between the two platforms (check gameplay at Figure 8). It can be categorized as a survival game: the objective is to stay alive as long as possible, cleaning up the level area by killing all enemies and gathering score. As a survival mode game, the player needs to realize the importance of keeping the health of its character, and questions are introduced into the game through items. When an enemy is destroyed, an item may or may not drop. When the player gathers the item the game freezes and a question (previously provided by the teachers) is displayed on screen (Figure 9). By answering the question correctly the student receives a *health boost* which allows him to stay longer in the game. When the student doesn't get the answer correctly, no health boost is given. This behaviour forces the student to keep gathering items and answering questions correctly, which allows him to go further in the game.

Figure 8: Overview of the gameplay.

The score system is made to incentivize competition amongst students: answering questions correctly gives the students far more points than killing enemies; while a mistaken answer will not provide any score points. With this approach, students are forced to focus their efforts into answering questions correctly but still engage into killing more enemies and keeping themselves alive, since the longer they play, the greater the score.

The game also provides in the GUI, information's about the game level, number of living enemies to be faced, current score, life-bar and reloading-bar and a mini-map. Arrow keys are used to move the character around, while mouse is used to point, aim and shoot into enemies.

Figure 9: Overview of the in-game questions provided by teacher's web platform (question used as an example).

Conclusion

Little has been done over the past years in Picos, Piaui, to produce educational material for computers. There's growing need for applications that may improve the usage of computer-labs in the educational system. This article has proposed a viable alternative to the "status quo", and the results gathered on mid-stage development are already positive, although many improvements are still being developed.

The results presented in section 4 are satisfactory, giving the early stage of development, but future work is much needed. Studies will be conducted to adapt the content of the game to fit age-level according to pedagogic monitoring, aiming to become a more accurate tool respecting the principles of pedagogy. Other improvements include the game's main menu, instructions, effective high-score systems and GUI. Future testing with a targeted audience will provide a more adaptable level development and difficulty adjusts. Further tests will evaluate the performance of HTML5 (fps, processor usage); the platform and how it contributes to improve the education-levels (by comparing results of different classes that have and have not used the software); and usability (how the users evaluate the software).

With positive results, this tool could provide a legacy to public schools in Picos; an alternative platform, free of charges that improves the educational levels of the city, while addresses the need for computer technologies in them. An investment in the future of those who benefit from these technologies, the children.

Bibliography

[1] BARBOSA, P.; MURALOLLI, P. Jogos e Novas Tecnologias na Educação. **Perspectivas em Ciências Tecnológicas**, p. 39-48, 2013.

[2] SAVI, R.; ULBRICHT, V. R. Jogos Digitais Educacionais: Benefícios e Desafios. **Revista Novas Tecnologias na Educação (RENOTE)**, Porto Alegre, v. 6.

[3] MINISTÉRIO DA EDUCAÇÃO. IDEB Picos. **Indice de Desenvolvimento da Educação Básica**, 2011. Available at : <http://www.portalideb.com.br/cidade/4770-picos/ideb>. Acessed in September 10, 2014.

[4] RUBY ON RAILS FRAMEWORK. **Ruby on Rails**, 2014. Available at: <http://rubyonrails.org/>. Acessed in September 11, 2014.

[5] BRUCE, T.; HIBBS, C. **Ruby on Rails:** Up and Running. O'Reilly Media, Inc, 2006.

[6] SUEHRING, S. **MySQL Bible.** Wiley Publishing, 2002.

[7] PILGRIM, M. **HTML5:** Up and Running. O'Reilly Media, Inc, 2010.

[8] SCIRRA. Construct 2, 2014. Available at: <https://www.scirra.com/construct2>. Accessed in September 10, 2014

[9] GARETT, J. Ajax: A New Approach to Web Applications. **Adaptive Path**, February 2005.

[10] INTRODUCING Json, 2014. Available at: <http://json.org/>. Accesed in September 10, 2014.

Uma arquitetura de balanceamento de carga web escalável para nuvens Eucalyptus

Pedro Roger Magalhães Vasconcelos [1]
Gisele Azevedo de Araújo Freitas [1]

Resumo: Computação em nuvem provê acesso a um conjunto de recursos como máquinas virtuais, armazenamento e rede como serviços. Neste contexto, o balanceamento de carga é importante para computação em nuvem, pois, permite a distribuição de carga de trabalho entre vários nós de processamento que podem ser provisionados sob demanda. Este artigo descreve um modelo para o balanceamento de carga web dinâmico com base em limites sobre o estado de execução dos nós, monitorados e escalados dinamicamente em função dos recursos: uso do processador, uso de memória e número de conexões. O trabalho demonstrou os benefícios do balanceamento de carga na Nuvem que é capaz de lidar com cargas repentinas, entregando recursos sob demanda e mantendo melhor utilização de recursos e infra-estrutura.

***Palavras-chave*:** Balanceamento de carga. Computação em nuvem. Escalabilidade.

Abstract: *Cloud computing provides access to a set of resources such as virtual machines, storage and network as services. In this context, load balancing is important for cloud computing because it allows the distribution of workload across multiple processing nodes that can be provisioned on demand. This article describes a model for dynamic web load balancing based on limits of the state of execution of the nodes, monitored and dynamically scaled according to resources: processor usage, memory usage and number of connections. The work has demonstrated the benefits of load balancing in the Cloud that is able to handle sudden loads, delivering on-demand resources and maintaining better resource and infrastructure utilization.*

***Keywords*:** *Cloud computing. Elasticity. Scalability.*

1 Introdução

A computação em nuvem emergiu como um novo paradigma de computação distribuída em larga escala que possibilitou transferir o poder de computação e os dados dos desktops e dispositivos móveis para os grandes datacenters [1]. Possui a capacidade de aproveitar o poder da Internet para a utilização de recursos disponíveis remotamente, provendo soluções rentáveis para a maioria dos requisitos reais [2]. É um modelo para permitir um acesso ubíquo, prático e sob demanda por rede a um conjunto compartilhado de de recursos computacionais (como redes, servidores, armazenamento e serviços) que podem ser rapidamente provisionados e liberados com um esforço mínimo de gerenciamento ou interação com o provedor de serviços.

A computação em nuvem é composta por cinco características essenciais, três modelos de serviço e quatro modelos de implantação [3]. Uma das características é a elasticidade, na qual os recursos são provisionados em várias formas e quantidades provendo escalabilidade aos sistemas. Aos usuários, os recursos aparentam serem ilimitados.

O balanceamento de carga é um mecanismo que distribui o excesso de carga de trabalho uniformemente entre todos os nós. É utilizada para alcançar uma alta satisfação de usuários e melhor taxa de utilização dos recursos, certificando-se que nenhum nó esteja sobrecarregado e contribuindo para a melhora do desempenho global do sistema [4]. Técnicas de balanceamento de carga podem prover utilização ótima dos recursos disponíveis, proteção a falhas, escalabilidade, redução tempo de resposta, além de evitar gargalos e excesso de provisionamento [5].

[1]Programa de Pós-Graduação em Engenharia Elétrica e de Computação), UFC, Campus Sobral - Sobral (CE) - Brasil
{pedro.roger@alu.ufc.br, gisele@lia.ufc.br}

A Computação utilitária pode ser definida como o provisionamento de recursos de computação e armazenamento como um serviço medido, semelhantes aos serviços públicos tradicionais como eletricidade, telefonia e água [6]. A cobrança por tais serviços é feita baseada no total de recursos utilizados ao invés de uma taxa fixa. Ao adotar tais modelos de provimento de serviços, também chamados pay-as-you-go, as empresas são capazes de evitar um grande investimento inicial, pagando apenas o custo de operação de suas funcionalidades [7].

Este trabalho propõe uma arquitetura de balanceamento de carga de aplicações web executada em uma nuvem privada Eucalyptus, na qual os estados das instâncias servidoras de aplicações e a utilização de recursos são constantemente monitorados. Este monitoramento é feito por um sofware que baseado nos dados obtidos pode otimizar o sistema iniciando novas instâncias, caso as instâncias servidoras estejam sobrecarregadas, ou removendo instâncias que estejam ociosas.

O Eucalyptus [8] (um acrônimo para Elastic Utility Computing Architecture for Linking Your Program To Useful Systems) é um framework de código aberto para a implantação de nuvens privadas de infraestrutura. Ele permite que os seus usuários executem e controlem máquinas virtuais utilizando uma dada infraestrutura física. Eucalyptus provê uma API baseada na Amazon EC2 [9] e Amazon S3 [10], permitindo compatibilidade com esses importantes serviços [11].

A plataforma Eucalyptus possui uma arquitetura modular e hierárquica, onde cada módulo é implementado como um serviço web a fim de alcançar mais compatibilidade e segurança. Os seis módulos que compõem a plataforma Eucalyptus são: Cloud Controller (CLC), Scalable Object Storage (SOS), Cluster Controller (CC), Storage Controller (SC), Node Controller (NC) e um módulo opcional chamado VMware Broker (VB) [12].

O monitoramento das instâncias é realizado através do sistema de monitoramento de código aberto Zabbix [13]. Essa ferramenta é um sistema de nível coorporativo designado para monitorar a disponibilidade e o desempenho de infraestruturas computacionais e de telecomunicações. Um agente Zabbix é instalado em cada instância e realiza o monitoramento dos recursos de modo passivo. Através de uma API o software controlador desenvolvido resgata o estado de utilização dos recursos provisionados a cada instância.

O balanceamento de carga das requisições é realizado com o servidor web Nginx [14], que é um servidor web leve e flexível que também pode atuar como proxy reverso, balanceador de carga e servidor de cache.

O software controlador mede constantemente o uso de três recursos das instâncias: utilização do processador, consumo de memória e quantidade de conexões.

Testes de desempenho foram realizados com o software Apache JMeter [15], nos quais uma carga de trabalho foi gerada para uma aplicação PHP servida na nuvem. Durante o teste, o sistema de monitoramento detectou uma sobrecarga nas instâncias iniciais e baseado nas métricas de monitoramento foi ajustando os recursos alocados para o processamento das requisições. O trabalho analisa o desempenho total do cluster quando monitorado sob cada uma das métricas citadas.

Este trabalho está organizado da maneira a seguir: A seção 2 apresenta uma fundamentação teórica necessária ao entendimento do trabalho. A seção 3 descreve trabalhos relacionados com o tema. A seção 4 apresenta e detalha a arquitetura de balanceamento de carga proposta. A seção 5 descreve os experimentos realizados e a análise dos resultados. Por fim, a conclusão do trabalho e as perspectivas futuras são descritas na seção 6.

2 Balanceamento de Carga e Computação em Nuvem

O balanceamento de carga é o processo de melhorar o desempenho de um sistema paralelo ou distribuído através da redistribuição de carga entre os processadores [16].

Alguns dos objetivos de um algoritmo de balanceamento de carga são [17]:

- Alcançar uma melhoria no desempenho global do sistema a um custo razoável.
- Tratar todas as requisições ao sistema de forma igualitária independente de sua origem.
- Possuir tolerância a falhas.

- Possuir a habilidade de modificar-se de acordo com qualquer alterações ou expandir a configuração do sistema distribuído.
- Manter a estabilidade do sistema.

O balanceamento de carga dinâmico pode ser realizado em abordagens diferentes: distribuído e não distribuído. Na abordagem distribuída, o algoritmo de balanceamento é executado em todos os nós presentes no sistema e a tarefa de balanceamento é distribuída entre eles [18]. Este tipo de balanceamento geralmente utiliza mais mensagens que os não distribuídos, pelo fato de que cada nó do sistema necessita interagir com todos os outros nós. Um benefício dessa abordagem é que mesmo que um ou mais nós do sistema fiquem indisponíveis, não ocorrerá o comprometimento total do processo de balanceamento, porém, poderá haver degradação de desempenho. O balanceamento não distribuído pode possuir duas formas: centralizado e não centralizado. Na primeira, o algoritmo de balanceamento de carga é executado em um único nó do sistema (nó central) e requer menos mensagens para alcançar a decisão de balanceamento. Isso acontece pelo fato de que os outros nós no sistema não interagem uns com os outros, eles apenas interagem com o nó central. Por outro lado, algoritmos de balancemento centralizado colocam em perigo o desempenho do sistema caso ocorra indisponibilidade do nó central. Há também a possibilidade de que esse nó cause um gargalo se ele for congestionado com mensagens de todos os outros nós do sistema[18]. Um estudo realizado por tem demonstrado que o balanceamento de carga centralizado é mais adequado a sistema de pequeno porta (menos de 100 nós) do que qualquer outro método de controle.

Nuvens de Infraestrutura como Serviço (IaaS) devem provisionar dinamicamente seus recursos de modo a aumentar a escalabilidade dos sistemas em momentos de pico e diminuir a escalabilidade nos momentos de subutilização, a fim de maximizar a eficiência de uso de recursos e/ou minimizar os custos associados. A maioria dos autores concorda que uma nuvem consiste de aglomerados de computadores distribuídos que fornecem recursos ou serviços por demanda a uma rede com a escala e confiabilidade de um centro de dados [19]. Noções de virtualização de recursos e grade de computação podem ser utilizadas para que estes aglomerados forneçam instâncias por demanda.

Atualmente, combinar a tecnologia de virtualização com a nuvem computacional tem sido uma tendência. Um plataforma em nuvem é um enorme recipiente que pode mesclar diferentes tecnologias de virtualização e aplicações em conjunto através da Internet. Considerando o elevado custo na gerência de computadores físicos ou servidores, cada vez mais pessoas e empresas voltam sua atenção para os serviços em nuvem que podem ser oferecidos através da Internet. Os usuários finais tendem a usar recursos virtuais para processamento de dados em larga escala, computação, renderização de animações e armazenamento de dados. Estes serviços dependem de redes de altas taxas de transmissão, grande capacidade de processamento e armazenamento, projetos de interações mais adequados e alta disponibilidade de serviços. Prover serviços de segurança consistentes em nuvens de infraestrutura é de fundamental importância devido à natureza multi-locatário e multi-fornecedor das plataformas em nuvem [20].

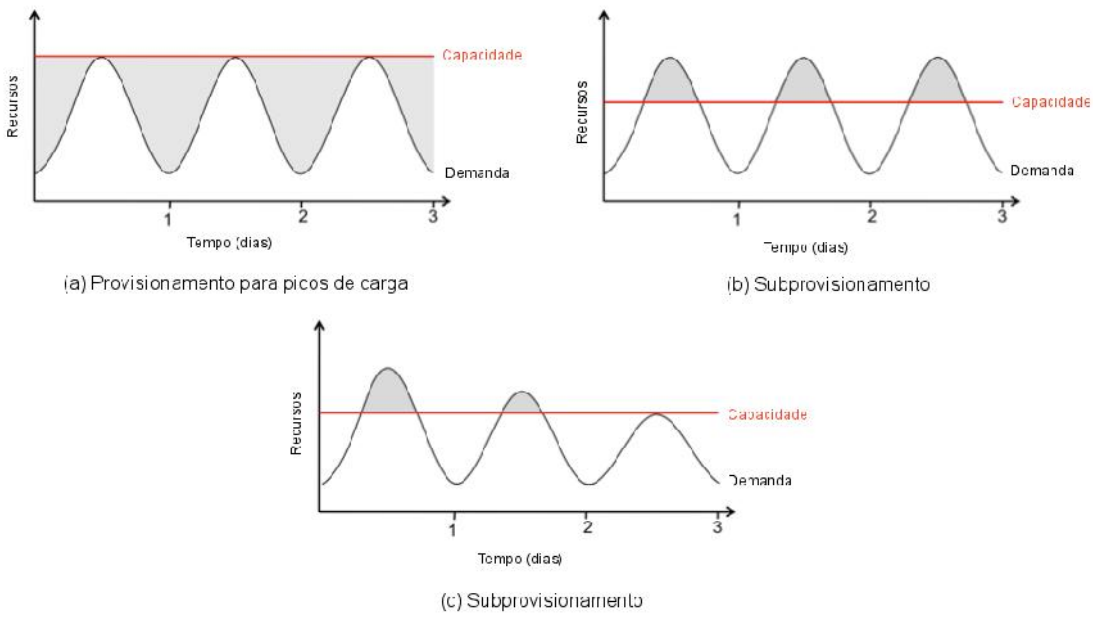
Portanto, a combinação de plataforma de infraestrutura de nuvem e virtualização é um ponto-chave na prestação de serviços de clientes. Com o objetivo de oferecer aos clientes uma experiência eficiente e confiável quando eles exigem uma máquina virtual a partir da internet, precisam do apoio de instalações de hardware para oferecer uma infraestrutura essencial, uma velocidade alta e ambiente de rede segura para construir uma conexão entre clientes e serviços em nuvem, um conjunto de recursos disponíveis para fornecer recursos virtuais reais.

Várias organizações estão migrando para serviços de computação em nuvem para diminuição de riscos e melhor disponibilidade dos negócios. Em acessos sob demanda os usuários requisitam serviços de infraestrutura para acesso imediato e por um curto intervalo de tempo, a cobrança dos serviços é feita baseada nessa duração. As requisições são servidas pela alocação de máquinas virtuais servidoras de aplicações que são alocadas no hardware subjacente [21].

A escalabilidade permite que os usuários provisionem recursos suficientes para os períodos de picos de carga e evitem a subutilização de recursos em momentos de não pico [1]. A Figura 1(a) ilustra que mesmo com a antecipação do provisionamento para a carga de pico, sem a elasticidade há desperdício de recursos (área sombreada) durante períodos de não pico. A Figura 1(b) ilustra o efeito do subprovisionamento quando o sistema é submetido a cargas de pico e a Figura 1(c) ilustra outro efeito do subprovisionamento de recursos que é o comprometimento do experiência dos usuários ao acessarem os sistemas. Os usuários abandonam o site permanentemente

após experenciarem mau serviço [1].

Figura 1: Provisionamento para picos de carga e subprovisionamento de recursos [1]



3 Trabalhos relacionados

Balanceamento de carga é um tema de frequentes trabalhos de pesquisa que têm como objetivo uma melhor distribuição dos recursos de forma a melhor atender suas cargas de trabalho.

As técnicas de balanceamento de carga em nuvens existentes, consideram vários parâmetros como desempenho, tempo de resposta, escalabilidade, utilização de recursos, tolerância a falhas, tempo de migração, sobrecargas, consumo de energia e emissão de carbono[5].

Katyal [22] discutiu vários esquemas de balanceamento de carga, concluindo que o balanceamento de carga estático possui ambientes de simulação e monitoramento mais simples mas falham na modelagem da natureza heterogênea da nuvem. Por outro lado, o balanceamento de carga dinâmico é difícil de simular mas é melhor adaptado aos ambientes heterogêneos de computação em nuvem.

Em [23] os autores apresentam uma estratégia de balanceamento de carga dada por um algoritmo genético que mantém um histórico do estado de execução das instâncias, conseguindo uma predição da influência que a ação de escalonamento terá no sistema, escolhendo a ação que alcance o melhor balanceamento de carga e reduza a flutuação do escalonamento. Essa estratégia resolveu o problema de desbalanceamento de carga e alto custo de migração dos algoritmos tradicionais.

[24] propuseram uma arquitetura de escalabilidade baseada em um monitoramento simples do estado das máquinas virtuais através de túneis SSH, a fim de obter um modelo de criação de instâncias de custos otimizados. Eles desenvolveram um servidor de monitoramento que possui dois componentes, um componente central que coleta o estado de execução das instâncias e realiza comandos de escalabilidade no controlador da nuvem, além de um front-end web para interação do administrador e geração de gráficos.

Em [25] foi proposta uma arquitetura que utiliza computação em nuvem em conjunto com princípios de computação autonômica para o provimento de um ambiente elástico. Nos experimentos foram utilizadas uma nuvem privada OpenNebula, cargas de trabalho com o software HTTPERF e um sistema de multiplicação de matrizes em Java. A métrica utilizada para disparar ações de elasticidade foi a média do percentual de utilização dos processadores das máquinas virtuais.

Os autores de [21] propuseram um modelo de alocação eficiente de máquinas virtuais a fim de diminuir o tempo de alocação e otimizar a utilização de recursos para acesso sob demanda em uma nuvem OpenNebula. Os seus experimentos mostraram que o algoritmo proposto pode melhorar a utilização de recursos para servir um maior número de requisições de recursos sem comprometer o tempo de alocação. Quando uma requisição de máquina virtual chega ao agendador, ele calcula a razão entre a especificação da máquina virtual requisitada pela especificação dos host físico. Esse valor é calculado para cada um dos hosts físicos que estão organizados em uma árvore de pesquisa binária começando do nó raiz até os nós folha, até que se encontre o melhor ajuste.

4 Arquitetura do modelo de balanceamento de carga

Eucalyptus possui uma arquitetura modular e hierárquica onde cada módulo é implementado como o serviço web a fim de alcançar mais compatibilidade e segurança. Os seis módulos que compõem a plataforma Eucalyptus são: Cloud Controller (CLC), Scalable Object Storage (SOS), Cluster Controller (CC), Storage Controller (SC), Node Controller (NC) e um módulo opcional chamado VMware Broker (VB) (Figura 2) [12].

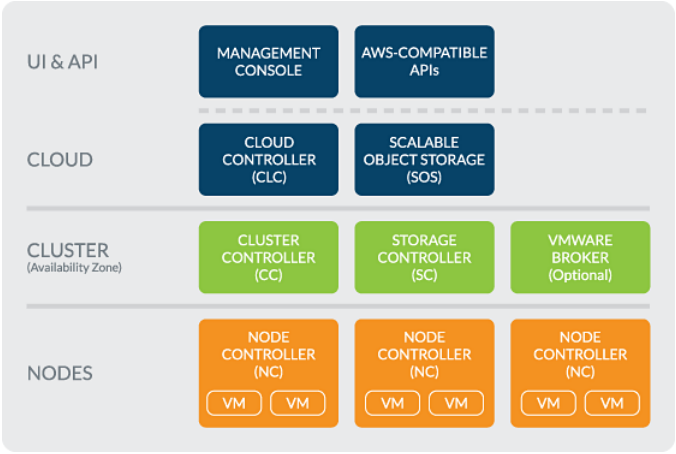


Figura 2: Componentes Eucalyptus.

- **Cloud Controler:** O Cloud Controler (CLC) é o ponto de entrada da nuvem para os seus administradores, desenvolvedores e usuários finais. O CLC consulta informações de outros componentes, realiza decisões de agendamento e realiza requisições ao Cluster Controler. Como uma interface para a plataforma de gerenciamento, o CLC é responsável por expor e gerenciar os recursos virtualizados subjacentes (servidores, rede e armazenamento). Apenas um CLC pode existir por nuvem.

- **Scalable Object Storage:** É um serviço Eucalyptus similar ao AWS Simple Storage Service (S3). O Scalable Object Storage (SOS) é um serviço que permite aos administradores da infraestrutura terem a flexibilidade de implementar armazenamento escalável para nuvens de larga escala. Eucalyptus também provê um sistema de armazenamento básico, chamado Walrus, mais adequado para avaliações e nuvens de pequena escala.

- **Cluster Controller:** Um cluster é equivalente a uma zona de disponibilidade AWS, e um uma nuvem Eucalyptus pode possuir vários clusters. O Cluster Controller (CC) age como um *front end* para um cluster dentro de uma nuvem Eucalyptus e se comunica com o Storage Controller e o Node Controller. O CC gerencia a execução de instâncias e Acordos de Níveis de Serviço (SLA) por cluster.

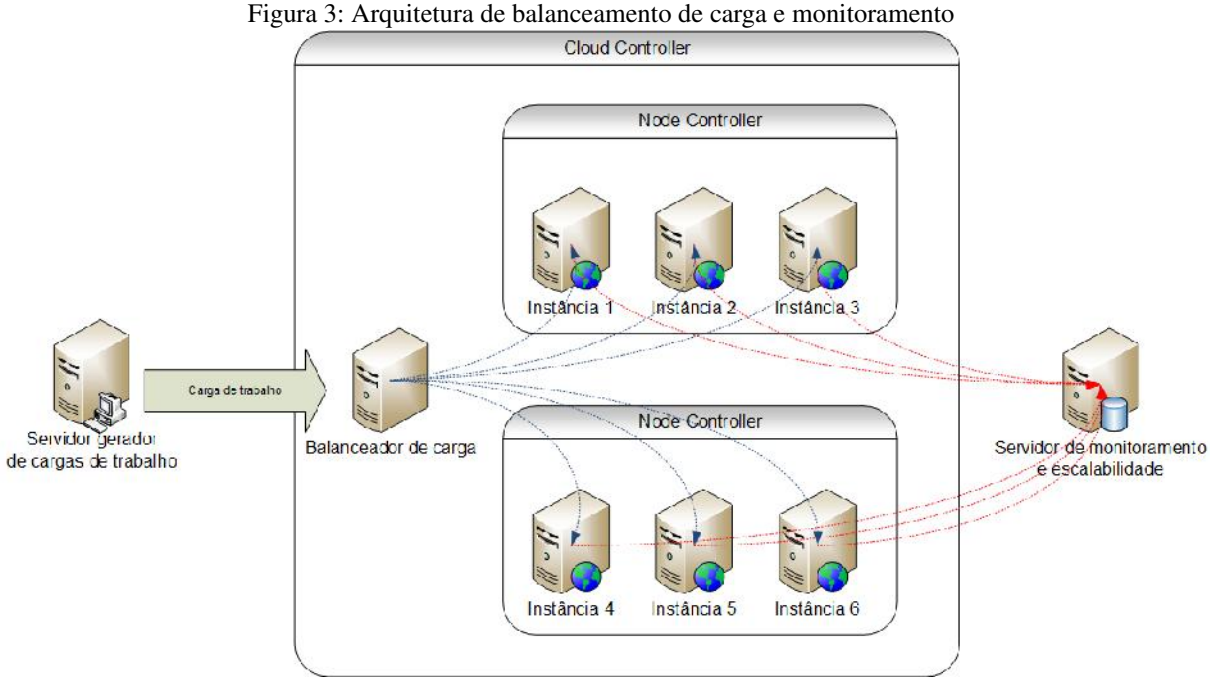
- **Storage Controller:** É um componente escrito em java e é equivalente ao AWS Elastic Block Store (EBS). O Storage Controller (SC) realiza comunicações com o Cluster Controller e Node Controller e gerencia volumes e *snapshots* de dispositivos de bloco para as instâncias dentro de um cluster. O SC realiza interfaces com vários sistemas de armazenamento, como NFS, iSCSI e SAN.

- **VMware Broker:** É um módulo opcional que permite ao Eucalyptus lançar máquinas virtuais em uma infraestrutura VMware. O VMware Broker (VB) media toda a comunicação entre o Cluster Controler e os hipervisores VMware (ESX e ESXi).
- **Node Controller:** O Node Controller (NC) é executado nas máquinas que hospedam máquinas virtuais. O NC controla atividades das máquinas virtuais como execução, inspeção e término de instâncias. Ele também recupera e mantém um cache local de imagens de instâncias.

Uma instância é lançada com a finalidade de prover o balanceamento de carga das requisições. Essa instância executa o servidor Nginx que recebe as conexões vindas do exterior da nuvem e balanceia as requisições de forma igualitária, utilizando um algoritmo Round-robin, entre as instâncias servidoras.

O diagrama apresentado na Figura 3 ilustra a arquitetura de balanceamento.

Figura 3: Arquitetura de balanceamento de carga e monitoramento



Um script controlador foi desenvolvido e executado no servidor de monitoramento. Tal script utiliza conexões seguras, através da autenticação por chaves do protocolo SSH, para comunicar-se com o Cloud Controller. As instâncias são manipuladas e os dados obtidos através da execução de comandos do pacote de ferramentas eucatools, como os comandos euca-describe-instances, euca-run-instances e euca-terminate-instances. Inicialmente o controlador recupera a lista de instâncias servidoras em execução. A seguir passa a monitorar a utilização do recurso selecionado em tais instâncias e caso a média de utilização deste recurso ultrapasse determinado limiar superior, a ordem de iniciar uma nova instância é emitida ao Cloud Controller. Caso a utilização do recurso esteja abaixo de um limite inferior, significa a subutilização dos recursos provisionados e uma ordem de remoção de uma instância é emitida. Se a utilização destes recursos se encontrar entre os dois limites, significa que a carga total do sistema está aceitável e que ainda possui uma margem de recursos livres.

5 Experimentos e análise dos resultados

Para avaliar a arquitetura proposta um experimento foi realizado utilizando uma nuvem na plataforma Eucalyptus versão 3.4.2 executada em um sistema operacional CentOS Linux 6.5 64 bits. Os servidores utilizados foram duas IBM BladeCenter HS21 com dois processadores Intel Xeon E5-2620 de 2GHz (6 núcleos cada), 15MB

de cache L2 por núcleo e 48GB de RAM, conectadas através de uma rede Gigabit Ethernet. As instâcias utilizadas foram criadas a partir de imagens EMI do Debian 7 64 bits distribuídas pelo aplicativo eucastore do próprio Eucalyptus. Todas as instâncias são do tipo "m1.small"que aloca 1 VCPU, 256 MB de memória e 5 GB de disco. Uma aplicação PHP foi desenvolvida e instalada nas instâncias servidoras, nas quais rotinas são executadas a fim de fazer uso exaustivo de processador e memória quando sob alta concorrência. A aplicação calcula uma série de Fibonacci de tamanho variável e aloca uma quantidade de memória durante sua execução. A carga de trabalho é gerada por um servidor externo à nuvem, simulando clientes reais acessando o sistema através da Internet. Utilizamos o servidor web Nginx v1.6.0 como balanceador de carga e o aplicativo Apache JMeter v2.11 para simular uma concorrência de 100 usuários realizando 40 iterações cada, totalizando 4000 requisições ao script. O sistema de monitoramento Zabbix v2.0.5 e o software controlador estão instalados em uma instância na mesma Zona de Disponibilidade que as outras.

O software controlador da escalabilidade monitora continuamente o consumo dos recursos das instâncias e toma as decições baseadas nos dados obtidos conforme a Tabela 1.

Tabela 1: Métricas de escalabilidade

Métrica	Escala para cima	Escala para baixo
Utilização de CPU	$> 80\%$	$< 30\%$
Utilização de memória	$> 60\%$	$< 40\%$
Número de conexões	> 80	< 30

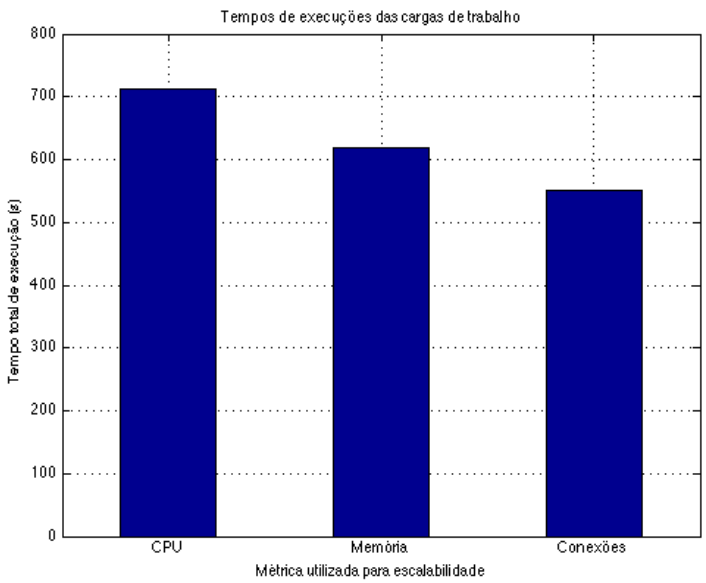
O intervalo entre as coletas de dados de monitoramento foi de 5 segundos e o período de espera antes de se enviar requisições a uma nova instância foi de 60 segundos. Esse intervalo é necessário devido ao tempo gasto para o provisionamento dos recursos da instância bem como seu processo de boot.

5.1 Análise dos resultados

O objetivo do experimento é observar o desempenho do cluster quando monitorado sob diferentes métricas. Assim, pode-se observar se a criação e destruição de novas instâncias ocorre com mais brevidade e menos flutuações quando sob efeito do monitoramento de uma ou outra métrica.

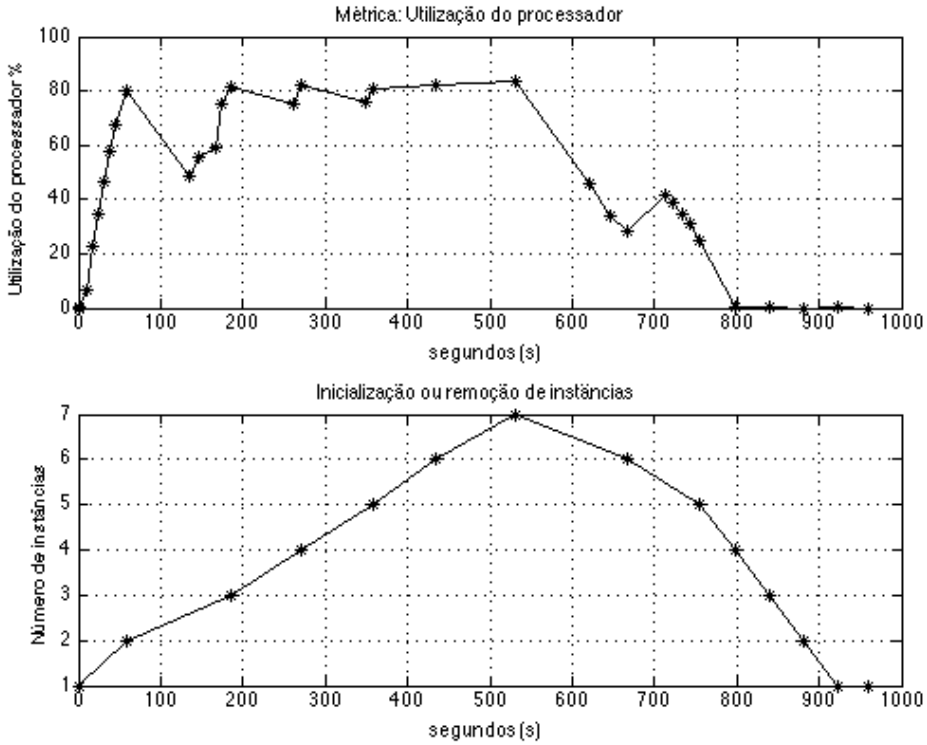
A Figura 4 demonstra o tempo total de execução das requisições sob cada métrica de escalabilidade.

Figura 4: Tempo de execução dos testes por cada métrica de escalabilidade



A Figura 5 demonstra a média de utilização do processador de todas as instâncias em execução e o comportamento da escalabilidade.

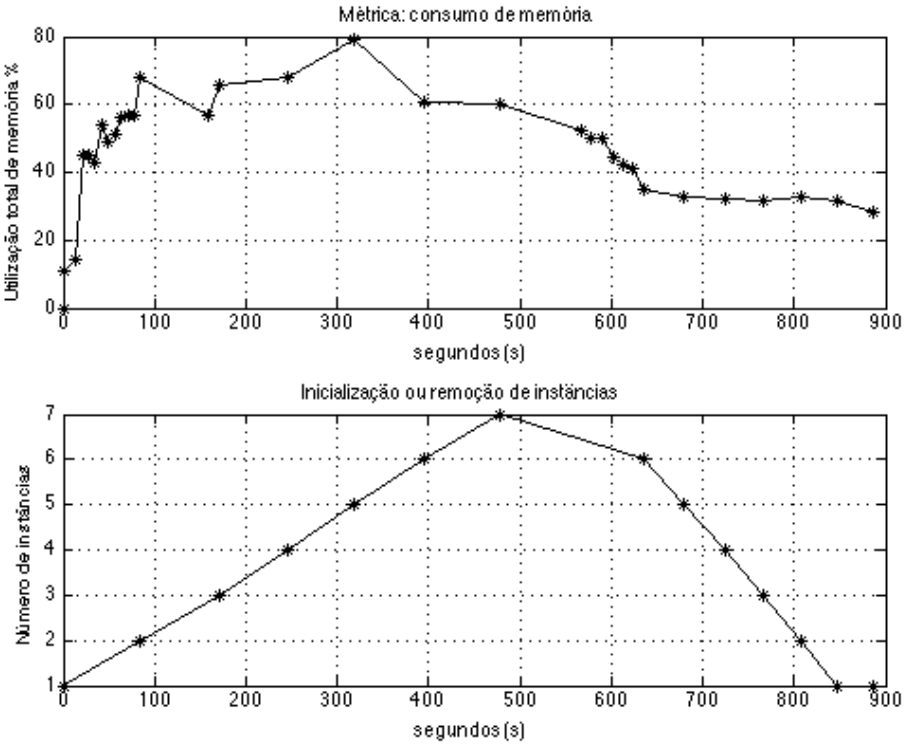
Figura 5: Utilização do processador



A carga do processador aumente a medida que novas requisições são enviadas às instâncias pelo balanceador de carga. Quando essa carga ultrapassa o limite superior uma ordem para a execução de uma nova instância é emitida pelo controlador para o Cloud Controller. Quando a nova instância entra em execução e novas requisições são enviadas a ela a utilização global de CPU tende a diminuir. Conforme a concorrência diminui, e consequentemente o uso do processador, o controlador ordena o término de instâncias a fim de manter o seu número compatível com a carga de trabalho corrente.

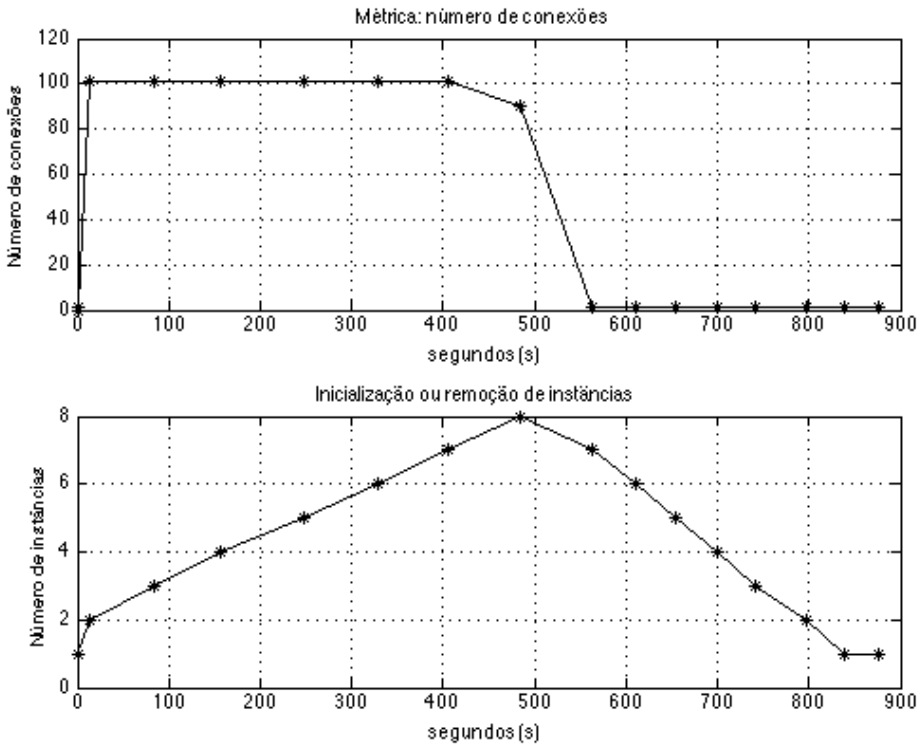
Tais comportamentos são válidos também quando utilizamos as outras métricas de balanceamento. A Figura 6 demostra o comportamento da escalabilidade baseada no consumo de memória e a Figura 7 relaciona a escalabilidade pela quantidade de conexões que ingressam no balanceador de carga.

Figura 6: Consumo de memória



O gráfico da escalabilidade por número de conexões, Figura 7, mostrou o menor tempo de execução, a maior quantidade de instâncias executadas e o menor tempo de inicialização entre instâncias. Esse fato ocorreu pela característica do teste utilizado, utilizamos a concorrência de 100 conexões e o limite superior da escalabilidade foi configurado como 80. Dessa forma, assim que o balanceador encaminha as 100 primeiras conexões à única instância inicial o gatilho superior de 80 conexões é disparado e o sistema escala logo nos primeiros segundos do teste, o que justifica o seu melhor desempenho. De forma equivalente, os gatilhos para os parâmetros processador e memória só são disparados depois de alguns ciclos de monitoramento, devido ao comportamento do consumo de tais recursos ser mais progressivo.

Figura 7: Número de conexões



Como efeito do balanceamento de carga e da escalabilidade, nas Figura 8, 9 e 10, observamos que a quantidade de transações processadas por segundo aumenta à medida que mais instâncias são adicionadas ao sistema.

Figura 8: Transações por segundo - CPU

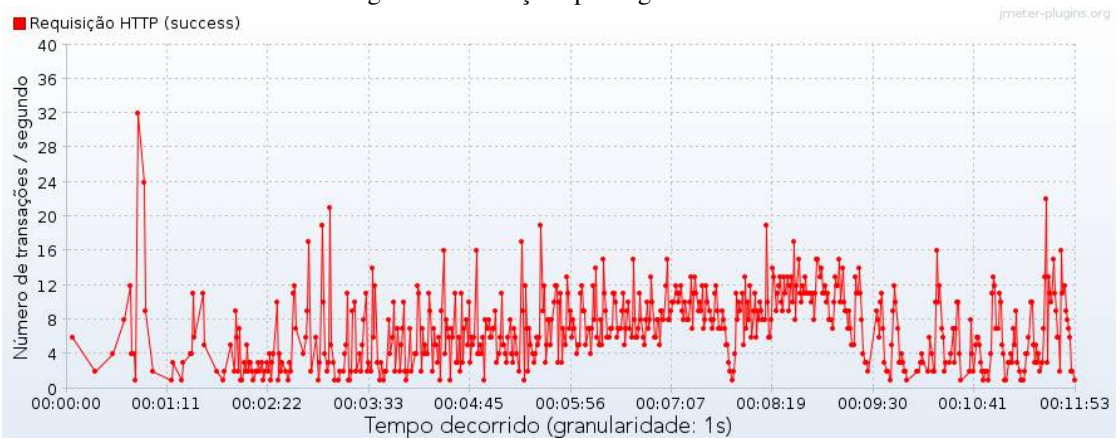


Figura 9: Transações por segundo - Memória

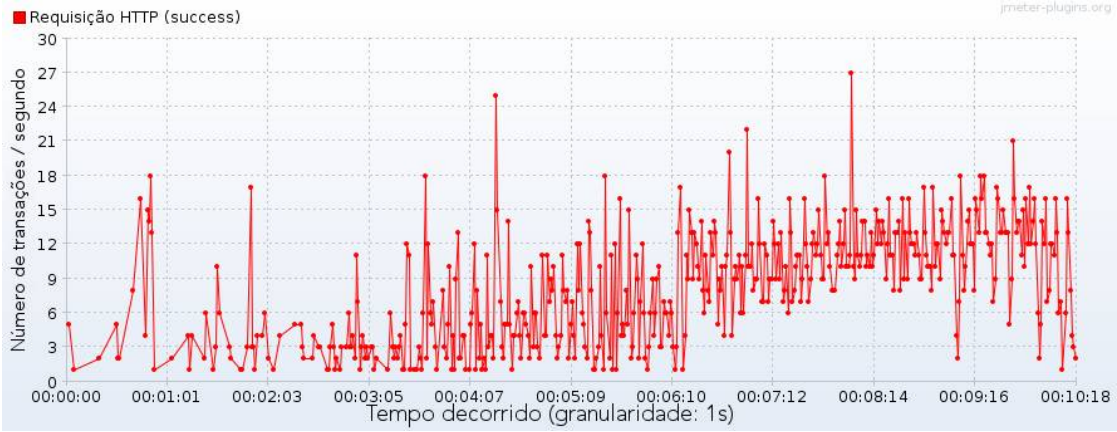
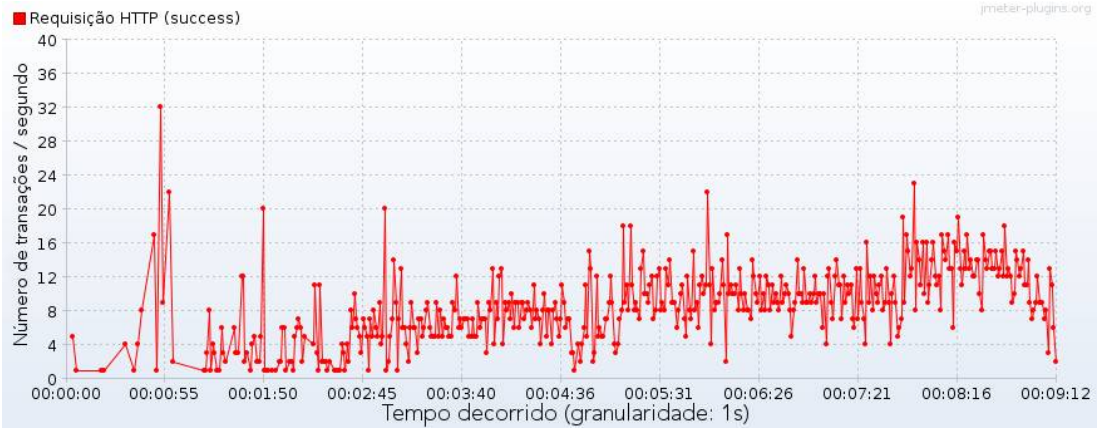


Figura 10: Transações por segundo - Número de conexões



6 Conclusão

Este trabalho apresenta um arquitetura de balanceamento de carga web para uma nuvem privada Eucalyptus que provê escalabilidade sob demanda baseada no estado das instâncias. Abordamos e discutimos os resultados da escalabilidade da nuvem em função do estado de três métricas: utilização dos processadores, consumo de memória e número de conexões.

O trabalho demonstrou alguns benefícios das plataformas de computação em nuvem, que são capazes de lidar com cargas repentinas, entregando recursos sob demanda para os usuários e mantendo maior utilização de recursos e infraestrutura, reduzindo assim, custos de gestão.

A maior contribuição do trabalho foi apresentar uma proposta de balanceamento de carga dinâmica que pode evitar problemas de subprovisionamento de recursos, e possíveis problemas de sobrecarga de sistemas e provisionamento para cargas de pico, o que poderia gerar recursos ociosos.

O software escalonador permite configurar limites superiores e inferiores que devem ser trabalhados baseados no tipo de aplicação servida na nuvem, se esta fizer uso intensivo de processamento ou memória. Ou ainda, se haverá um grande número de conexões ou tráfego em rajadas. Nos experimentos, a escalabilidade servida conseguiu provisionar os recursos necessários ao tratamento da carga de trabalho evitando o subprovisionamento de recursos e terminando instâncias à medida que a carga total do sistema diminuía. Técnicas melhoradas de predição e testes de desempenho com outras soluções de balanceamento de carga serão objetos de pesquisas futuras.

Agradecimentos

Gostaríamos de prestar agradecimentos ao Núcleo de Tecnologia da Informação da Universidade Estadual Vale do Acaraú (UVA) por permitir a utilização de sua infraestrutura computacional para a implantação, desenvolvimento e testes da arquitetura apresentada.

Referências

[1] ARMBRUST, M. et al. *Above the Clouds: A Berkeley View of Cloud Computing*. [S.l.], 2009. Disponível em: <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf>.

[2] BHADANI, A.; CHAUDHARY, S. Performance evaluation of web servers using central load balancing policy over virtual machines on cloud. In: *Proceedings of the Third Annual ACM Bangalore Conference*. New York, NY, USA: ACM, 2010. (COMPUTE '10), p. 16:1–16:4. ISBN 978-1-4503-0001-8. Disponível em: <http://doi.acm.org/10.1145/1754288.1754304>.

[3] MELL, P.; GRANCE, T. *The NIST Definition of Cloud Computing*. [S.l.], 2011.

[4] LAHA, J.; SATPATHY, R.; DEV, K. Load balancing techniques : Major challenges in cloud computing - a systematic review. In: *IJCSN International Journal of Computer Science and Network*. [S.l.: s.n.].

[5] KANSAL, N. J.; CHANA, I. Cloud load balancing techniques : A step towards green computing. In: *IJCSI International Journal of Computer Science Issues,*. [S.l.: s.n.].

[6] CLOUD Computing Implementation, Management and Security. [S.l.]: CRC Press, 2010.

[7] IRAKOZE, I. *Cloud-Based Mobile Applications*. B.S. Thesis, 2013.

[8] EUCALYPTUS. *Eucalyptus | Open Source Private Cloud Software*. 2014. <http://www.eucalyptus.com/>.

[9] EC2, A. *AWS | Amazon Elastic Compute Cloud (EC2)*. 2014. <http://aws.amazon.com/pt/ec2/>.

[10] S3, A. *AWS | Amazon Simple Storage Service (S3)*. 2014. <http://aws.amazon.com/pt/s3/>.

[11] FOLCH, A. *Interface development for Eucalyptus based cloud*. Dissertação (Mestrado) — Vilnius Gediminas Techinical University, 2011.

[12] EUCALYPTUS. *Eucalyptus Cloud Computing Architecture*. 2014. Disponível em: <https://www.eucalyptus.com/eucalyptus-cloud/iaas/architecture>.

[13] ZABBIX. *Homepage of Zabbix :: An Enterprise-class Open Source Distributed Monitoring Solution*. 2014. <http://www.zabbix.com/>.

[14] NGINX. *NGINX | High Performance Load Balancer, Web Server, & Reverse Proxy*. 2014. <http://nginx.com/>.

[15] JMETER, A. *Apache JMeter*. 2014. <http://jmeter.apache.org/>.

[16] SHARMA, S.; SINGH, S.; SHARMA, M. Performance analysis of load balancing algorithms. In: *World Academy of Science, Engineering and Technology*. [S.l.: s.n.].

[17] MADHU, K. M.; SHAH, S. M. Study on dynamic load balancing in distributed system. In: *International Journal of Engineering Research & Technology (IJERT)*. [S.l.: s.n.].

[18] ALAKEEL, A. M. A guide to dynamic load balancing in distributed computer systems. In: *IJCSNS International Journal of Computer Science and Network Security*. [S.l.: s.n.].

[19] GROSSMAN, R. L. The case for cloud computing. *IT Professional*, IEEE Computer Society, Los Alamitos, CA, USA, v. 11, n. 2, p. 23–27, 2009. ISSN 1520-9202. Disponível em: <http://www.computer.org/csdl/mags/it/2009/02/mit2009020023-abs.html>.

[20] DEMCHENKO, Y. et al. Security infrastructure for on-demand provisioned cloud infrastructure services. In: LAMBRINOUDAKIS, C.; RIZOMILIOTIS, P.; WLODARCZYK, T. W. (Ed.). *CloudCom*. IEEE, 2011. p. 255–263. ISBN 978-1-4673-0090-2. Disponível em: <http://dblp.uni-trier.de/db/conf/cloudcom/cloudcom2011.html>.

[21] MANDAL, S. K.; KHILAR, P. M. Efficient virtual machine placement for on-demand access to infrastructure resourcesin cloud computing. *International Journal of Computer Applications*, v. 68, n. 12, p. 6–11, April 2013. Full text available. Disponível em: <http://research.ijcaonline.org/volume68/number12/pxc3887101.pdf>.

[22] KATYAL, M.; MISHRA, A. A comparative study of load balancing algorithms in cloud computing environment. *International Journal of Distributed and Cloud Computing*, 2013.

[23] HU, J. et al. A scheduling strategy on load balancing of virtual machine resources in cloud computing environment. In: *Proceedings of the 2010 3rd International Symposium on Parallel Architectures, Algorithms and Programming*. Washington, DC, USA: IEEE Computer Society, 2010. (PAAP '10), p. 89–96. ISBN 978-0-7695-4312-3. Disponível em: <http://dx.doi.org/10.1109/PAAP.2010.65>.

[24] KARMAKAR, K.; MANDAL, A. Cost optimized virtual machine deployment in eucalyptus for autonomous systems. *International Journal of Innovations in Engineering and Technology (IJIET)*, v. 3, n. 4, p. 6–11, April 2014. Disponível em: <http://ijiet.com/issues/volume-3-issue-4-april-2014/>.

[25] COUTINHO, E.; GOMES, D. G.; SOUZA, J. de. Uma proposta de arquitetura autonômica para elasticidade em computação em nuvem. In: *Proceedings of 4º Workshop de Sistemas Distribuídos Autonômicos*. Santa Catarina, Brasil: [s.n.], 2014. (WoSiDA 2014), p. 3–6. ISSN 2177-496X. Disponível em: <http://sbrc2014.ufsc.br/anais/files/wosida/anaisWoSiDA2014.pdf>.

Análise de Sentimentos de *tweets* nos dias de jogos da Seleção Brasileira de Futebol na Copa do Mundo da FIFA Brasil 2014 utilizando Mineração de Textos

José Adail Carvalho Filho[1]
João Lucas Araújo Leite[1]
Ticiana Linhares Coelho da Silva[2]

Resumo: O aumento das redes sociais nos últimos anos permitiu aos usuários se conectarem e compartilharem informações em tempo real, enviando-as a milhares de outros usuários em um curto espaço de tempo. Além disso, a maneira como os usuários interagem mudou. Os usuários de redes sociais costumam postar suas opiniões sobre os grandes eventos, lançamentos de produtos, catástrofes, epidemias, entre outros acontecimentos. Para acompanhar o que eles estão falando nas redes sociais pode ser um diferencial para as organizações que desejam elaborar melhores estratégias de marketing, obter *feedback* sobre algum produto ou determinado evento. No entanto, essa grande quantidade de dados ainda continua crescendo, e a análise desses dados de forma não automatizada pode ser um problema não trivial. Neste contexto, este artigo mostra como o processo de mineração de textos foi usado para coletar, estruturar o texto extraído do *Twitter* (*tweets*) e como criar um modelo de classificação de texto que permita predizer a opinião da rede social do usuário do Twitter sobre Copa do Mundo da FIFA Brasil 2014. As postagens dos usuários, popularmente conhecido como *tweets*, são categorizadas neste trabalho como um sentimento: positivo, negativo, ambíguas ou neutras.

Palavras-chave: Análise de sentimentos. Mineração de textos. Redes sociais.

Abstract: *The increase of the social networks in the last years allowed users to get connected and share information in real time, spreading it for thousands of others users in a short time. Also, the users interaction have changed. The users of social networks usually post their opinions about big events, product releases, catastrophes, epidemics, among other happenings. To follow what they are talking on the social networks may be a differential for organizations who wants to elaborate better marketing strategies, to obtain feedback about some product or a certain event, among other possibilities. Although, this big amount of data still keeps growing, and analysis it in a non-automated way may be a non-trivial problem. In this context, this article shows how the Text Mining process was used to collect, to structure the text extracted from Twitter(tweets) and to create a text classification model that allowed to predict the Twitter social network user's opinion about the FIFA World Cup Brazil 2014. The user's posts called as tweets are categorized in this work as a sentiment: positive, negative, ambiguous or neutral.*

Keywords: *Sentiment analysis. Social networks. Text mining.*

[1] Bacharelado em Engenharia de Software - UFC, Campus Quixadá - Av. José de Freitas Queiroz, 5003 – Cedro – Quixadá(CE) - Brasil
{adail.dux@gmail.com}{lucas.compufc@gmail.com}
[2] Professora Assistente – UFC, Campus Quixadá – Av. José de Freitas Queiroz, 5003 – Cedro – Quixadá (CE) - Brasil
{ticianalc@ufc.br}

1 Introdução

O rápido crescimento do uso da internet e a popularização das redes sociais mudou a forma de interação entre pessoas e organizações. Usuários publicam suas opiniões sobre as organizações, eventos, catástrofes, dentre outros, em seus perfis em redes sociais, que de maneira rápida são propagadas para vários outros usuários. Estas mensagens podem conter teores positivos, mas também podem ser duras críticas. Isso pode acarretar em vantagens para as organizações, mas também pode comprometer seriamente a imagem das mesmas, pela grande quantidade de usuários que atualmente as redes sociais possuem.

Essa popularização da internet, por sua vez, gera um grande volume de informação a cada instante, e as organizações, em geral, não conseguem acompanhar no mesmo ritmo o que os usuários estão comentando sobre as mesmas. No entanto, percebeu-se que ao analisar essas informações, as organizações poderiam ter a vantagem de conhecer as opiniões dos usuários sobre seus serviços ou produtos fornecidos a partir de dados das redes sociais. Neste contexto, a Mineração de Textos, também conhecida como Descoberta de Conhecimento em Texto, fornece um conjunto de técnicas que podem automatizar o processo de coleta e estruturação de informações contidas nos textos, e junto com a Análise de Sentimentos, as organizações podem se beneficiar dos resultados para os mais diversos fins, como elaborar estratégias de *marketing*, táticas de segurança, melhoria de serviços, dentre outros.

A Copa do Mundo ocorreu no Brasil neste ano. Ela é a maior competição internacional disputada pelas seleções de futebol masculinas principais das 208 federações afiliadas à FIFA[3]. Tendo em vista que foi um evento de grande cobertura e interesse social, cercado de polêmicas e manifestações devido ao seu grande custo, em contraste com outros problemas que a população brasileira julga mais importante[4], este trabalho teve como objetivo a coleta, estruturação e mineração de textos extraídos do Twitter, a fim de analisar o sentimento dos usuários acerca do evento, nos dias de jogos da Seleção Brasileira de Futebol.

Este trabalho está organizado da seguinte forma: a próxima seção apresenta os principais objetivos deste trabalho, a Seção 3 apresenta os principais conceitos e a Seção 4 alguns os trabalhos relacionados a este. Na Seção 5, a metodologia adotada neste trabalho é apresentada, bem como na Seção 6 os resultados obtidos. Na Seção 7 é apresentada a conclusão deste trabalho.

2 Objetivos

O objetivo principal deste trabalho é investigar o sentimento dos usuários do Twitter sobre a Copa do Mundo FIFA Basil 2014 nos dias de jogos da seleção brasileira de futebol, validando os resultados obtidos com acontecimentos ao longo dos jogos. Assim, pretende-se mostrar como as organizações podem utilizar a Mineração de Textos para analisar a opinião de usuários do Twitter, através dos *tweets*, mensagens de textos limitadas a 140 caracteres, de maneira automatizada, no intuito de extrair conhecimento relevante para que as mesmas acompanhem as discussões dos usuários na rede social, auxiliando as mesmas nos mais diversos planos coorporativos.

O segundo objetivo é gerar um modelo de classificação utilizando uma implementação do algoritmo Naive Bayes, que avalie a polaridade de um *tweet* (positivo, negativo neutro ou ambíguo) baseado em uma *hashtag*, palavra precedida pelo caractere #, utilizado pra marcar palavras-chave ou tópicos em um *tweet*. Esta classificação será realizada baseada no reflexo do sentimento expresso na *hashtag*. Este modelo é relevante, pois é capaz de classificar novos *tweets* sobre o tema em questão como positivo, negativo, neutro ou ambíguo.

Por fim, este trabalho apresenta nuvens de palavras, imagens compostas por palavras, para mostrar de maneira visual, a frequência da ocorrência das palavras em um dado texto: quanto maior for o número de ocorrências de uma palavra, maior a mesma será na nuvem de palavras. Assim, pode-se acompanhar a variação das palavras mais frequentes ao longo dos dias (foi criada uma nuvem de palavra para cada dia de jogo). Um exemplo de nuvem de palavras é apresentado na Figura 1 acima. Na ocasião, a nuvem foi gerada a partir dos textos coletados no dia de estréia do evento.

[3] http://pt.fifa.com/aboutfifa/worldcup/index.html
[4] http://jcrs.uol.com.br/site/noticia.php?codn=141091

Figura 1: Exemplo de nuvem de palavras

3 Referencial Teórico

A seguir, são apresentados os principais conceitos envolvidos no desenvolvimento deste trabalho.

3.1 Twitter

Com mais de 600 milhões de usuários cadastrados[5], o Twitter é uma rede social que permite a seus usuários postarem mensagens de texto rápidas, limitadas a 140 caracteres, conhecidas como *tweets*. Sua estrutura dinâmica permite que qualquer usuário tenha acesso às informações que são constantemente postadas, sem que para isso restrinja aos usuários possuírem alguma permissão de conexão entre eles. Esse é o grande diferencial do Twitter em relação a outras redes sociais populares, como o LinkedIn e o Facebook, tornando o Twitter uma rede social mais interessante de se explorar[1].

O Twitter fornece uma REST API para desenvolvedores, que permite aos mesmos acessarem atualizações, status, dados de usuários, etc. REST (*Representation State Transfer*) é uma arquitetura de redes de estilo híbrido derivada do estilo de várias arquiteturas baseadas em rede, que define uma interface conectora que permite aos clientes "conversar" com servidores de maneira única[6]. O Twitter também fornece aos desenvolvedores acesso a um grande volume de informações em tempo real através de uma *Streaming* API. A Twitter *Streaming* API fornece para desenvolvedores acessos de baixa latência para a *stream* global do Twitter de dados de *tweets*[7].

O Twitter será utilizado como fonte de exploração de dados para este trabalho, tendo em vista que o Twitter é uma rede social com uma grande quantidade de usuários ativos e possui alcance global, que induz seus usuários a compartilhar suas ideias constantemente, gerando grande quantidade de informação a cada instante, além de fornecer API's que permitem explorar dados dos tweets de maneira fácil.

3.2 Mineração de textos

A Mineração de Dados, também conhecida como Descoberta de Conhecimento em Base de Dados, é uma área que concentra seu trabalho na exploração automatizada em grandes massas de dados para descobrir padrões interessantes entre elas.

A Mineração de Textos, também chamada de Descoberta de Conhecimento em Texto, refere-se ao processo de extrair padrões interessantes e não triviais ou conhecimento a partir de textos desestruturados [2]. Nela, são aplicadas as mesmas funções analíticas da Mineração de Dados [3], porém para dados textuais. Os dados textuais englobam uma vasta e rica fonte de informação, mesmo em um formato que seja difícil de extrair de maneira automatizada [4]. Ela pode ser vista como uma extensão da Mineração de Dados.

[5] http://www.statisticbrain.com/twitter-statistics/
[6] http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
[7] https://dev.twitter.com/docs/api/streaming

Esta grande massa de informação textual desestruturada não pode ser utilizada por computadores para extração de conhecimento, uma vez que os mesmos a tratam apenas como uma sequência de caracteres. Assim, faz-se necessária a aplicação de diferentes métodos e algoritmos para dar estruturação aos dados textuais, visando facilitar a extração de conhecimento dos respectivos dados.

Embora a Mineração de Textos geralmente se refira à extração de conhecimento de base de dados textuais, é possível recorrer a outras áreas para facilitar a descoberta de conhecimento nos dados textuais. Assim, diversas técnicas e algoritmos podem ser utilizados para auxiliar o processo de extração de conhecimento dos textos. Deste modo, a Mineração de Textos pode estar ligada além das técnicas de aprendizagem de máquina e Estatística, a área de Processamento de Linguagem Natural, detalhada na subseção seguinte. Este trabalho utiliza Mineração de Textos para analisar as postagens sobre a Copa do Mundo 2014 realizadas na rede social *Twiiter*.

3.3 Pré-processamento de texto

No pré-processamento, podemos utilizar técnicas como o Processamento de Linguagem Natural para estruturarmos os textos que analisaremos.

O Processamento de Linguagem Natural é um conjunto de técnicas computacionais para analisar e representar ocorrências naturais de texto em um ou mais níveis de análise linguística com o objetivo de se alcançar um processamento de linguagem similar ao humano para uma série de tarefas ou aplicações [5].

Assim, o PLN naturalmente lida com diversos elementos linguísticos e estrutura gramatical, sendo um processo complexo, paralelo à complexidade linguística natural. Para processar a linguagem natural, o PLN a representa em diversos níveis, como léxico, morfológico, semântico, etc [5].

A Mineração de Textos, como citado anteriormente, se assemelha a Mineração de Dados, desde a definição do problema até a extração de conhecimento. Na Mineração de Dados, faz-se necessária na etapa de pré-processamento dos dados a aplicação de métodos para transformação, limpeza, seleção e redução de volume de dados, antes da etapa de mineração [6]. Assim, para esta etapa da Mineração de Textos, utilizamos alguns métodos para transformação e limpeza dos dados textuais, como uso de expressões regulares para extrairmos do texto padrões que não são relevantes para os resultados de classificação, como *links*, caracteres especiais, etc.

3.4 Análise de Sentimentos

A larga expansão da internet gera muitas informações em forma de opiniões em seus diversos canais: fóruns,comunidades, redes sociais, etc. As opiniões são tão importantes que onde quer que se queira tomar decisões as pessoas querem ouvir a opinião de outros [7]. Isso não é uma verdade apenas para as pessoas, como também para as organizações, afinal, conhecer a opinião dos clientes acerca dos seus produtos e serviços é de grande valia para as organizações.

A Análise de Sentimentos ou Mineração de Opinião é o estudo computacional de opiniões, sentimentos e emoções expressos em texto. A informação textual pode ser classificada em dois principais tipos: fatos e opiniões. Fatos são expressões objetivas sobre entidades, eventos e suas propriedades. Opiniões são geralmente expressões subjetivas que descrevem os sentimentos da população, avaliações, ou sentimentos em relação as entidades, e suas propriedades[7]. Ela é a área que ajuda de maneira automatizada a determinar a direção do sentimento em textos (positivo ou negativo). Apesar da Análise de Sentimentos ser apresentada por grande parte da literatura como estudo computacional de sentimentos, a mesma pode ser utilizada para muitos outros projetos [2]. Tendo em vista que a Análise de Sentimentos se trata de um problema de classificação, ela pode ser utilizada para classificar dados textuais, segundo sua polaridade, mesmo se o texto não denotar algum sentimento.

A Análise de Sentimentos é utilizada neste trabalho para classificar os textos dos *tweets* coletados de acordo com a sua polaridade (positiva ou negativa) de maneira automatizada. Essa classificação permitirá saber qual o sentimento expresso nas postagens dos usuários do Twitter referentes à Copa do Mundo 2014. A seguir, será apresentada a metodologia utilizada neste trabalho.

4 Trabalhos Relacionados

Tan [1] e Gomes [2] pontuam a relevância da aplicação de métodos de Mineração de Textos para a extração de conhecimento em base de dados textuais, uma vez que a Mineração de Dados é comumente aplicada

em dados que possuem certo nível de estruturação e contemplam apenas uma parte limitada de dados que as organizações possuem, ou seja, dados estruturados [2].

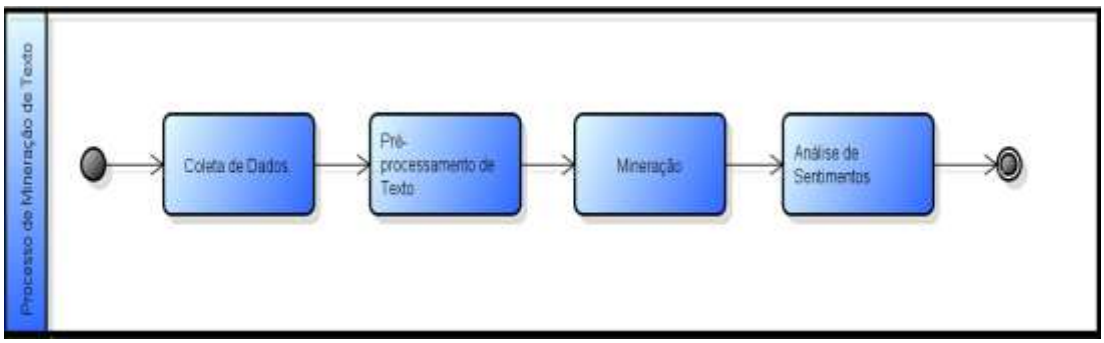
Gomes [2] aplica a Mineração de Textos em seu trabalho na busca de extrair conhecimento acerca de notícias de economia de Portugal. Assim, o autor visita sítios de notícias sobre economia de seu país para representar o sentimento expresso nas notícias, analisando os textos dos títulos das notícias. Barbosa et al [8] utilizam os processos da Mineração de Textos para explorar *tweets* que falam sobre as eleições presidenciais do Brasil do ano de 2010, a fim de traçar o sentimento online da população expresso nos *tweets* através das *hashtags*, classificando-os em positivos, negativos, neutros ou ambíguos, e correlacionar a classificação dos *tweets* aos fatos que ocorriam no Brasil relacionados a eleições, como debates políticos, por exemplo. Este trabalho assemelha-se ao de Barbosa et al [8], por utilizar *hashtags* para determinar o sentimento expresso em um *tweet,* correlacionando os resultados aos fatos ocorridos nos dias das partidas da Seleção Brasileira de Futebol. Entretanto, difere-se por considerar outras palavras do *tweet* que possam expressar um sentimento, mesmo que estas não estejam marcadas como uma *hashtag*.

Neste trabalho, foi aplicado o processo de Mineração de Textos semelhante ao trabalho de Gomes [2], no entanto a aplicação foi realizada em dados coletados na rede social Twitter. A escolha de usar essa rede social é pelo alcance global da mesma, que possui milhões de usuários cadastrados. Ao contrário de Gomes [2] que restringe o alcance do estudo a apenas a um lugar específico. Os textos minerados compõem um *tweet*, que é uma sequência de caracteres publicada pelos usuários, podendo conter outros tipos de dados anexados.

5 Metodologia

O trabalho foi dividido basicamente em 4 etapas, inspirado no processo de Mineração de Textos proposto por Aranha[9], que propõe que o processo de Mineração de Textos passe por cinco fases distintas. A primeira é a coleta dos dados textuais que serão minerados. A segunda é a etapa de pré-processamento dos dados, para dar uma melhor estrutura para que os mesmos possam ser submetidos aos algoritmos de mineração. A terceira fase consiste na criação de índices para fins de recuperação de dados. Na quarta ocorre a extração de conhecimento e na quinta ocorre a análise dos resultados obtidos. A figura 2 (abaixo) ilustra as etapas utilizadas nesse trabalho. A etapa de indexação foi descartada, por não ter sido considerada de relevância para este trabalho.

Figura 2: Processos de Mineração de Textos



5.1 Coleta de dados

A coleta de *tweets* que falavam sobre a Copa foi a primeira parte da execução deste trabalho, ocorrida nos 7 dias em que a Seleção Brasileira de Futebol disputou partidas: 12, 17, 23 e 28 de junho; 4,8 e 12 de julho. Para esta etapa, foi utilizado um *script* escrito na linguagem Python, para ter acesso a *Streaming* API do Twitter. Para a realização da coleta, foi definido um conjunto de *hashtags*, relacionadas à Copa do Mundo, passados ao script como parâmetros. Além disso, as postagens do Twitter foram monitoradas ao longo dos jogos para verificar quais *hashtags* relacionadas à Copa do Mundo estavam nos *trending topics*. *Trending topics* são os assuntos mais populares do momento no Twitter. Assim, à medida que *hashtags* relacionadas ao evento surgiam entre as mais comentadas, as mesmas eram repassadas ao *script* como parâmetro. Quando saiam dos *trending topics*, as mesmas eram removidas do script. A API retorna ao script *tweets* que contenham em seu texto a *hashtag* repassada. Esses *tweets* foram em seguida armazenados em um arquivo com valores separados por vírgula (*Comma Separated Values*), para que em seguida fossem submetidos ao pré-processamento.

5.2 Pré-processamento de texto

Neste trabalho, foram descartados conteúdos considerados irrelevantes para o processo de classificação de texto, como *links*, nomes de usuário (no Twitter, marcados pelo caractere '@') e caracteres não alfabéticos, com exceção do caractere '#', que é utilizado marcar uma palavra como *hashtag*.

Ainda nesta fase, foi aplicada a remoção de *stopwords*, uma técnica de Processamento de Linguagem Natural para a retirada de *stopwords*. Assim, palavras que não possuem relevância para os resultados da classificação de textos foram descartadas dos conjuntos de dados, melhorando o desempenho do algoritmo de classificação de textos que será utilizado e trazendo resultados mais relevantes.

Para este trabalho, também foi aplicado a técnica de PLN conhecida como remoção de *stopwords*. *Stopwords* são palavras que podem ser consideradas irrelevantes para um conjunto de resultados a ser exibido em uma busca realizada em uma *search engine*.[8] No contexto deste trabalho, as mesmas são irrelevantes por não serem palavras que expressam algum sentimento, podendo vir a atrapalhar na obtenção de bons resultados, por serem palavras que ocorrem com muita frequência nos textos. Alguns exemplos de *stopwords*: os, as, em, para, com, e.

5.3 Mineração de textos

Para esta etapa, foi utilizado o algoritmo de classificação de textos Naive Bayes da biblioteca de aprendizagem de máquina Apache Mahout, para gerar o modelo de classificação de *tweets*. O Mahout é uma biblioteca de aprendizagem de máquina de código aberto do Apache [10]. O Naive Bayes é um simples classificador probabilístico baseado na aplicação do teorema de Bayes (ver Figura 4) com uma forte independência de suposições[9]. Ele foi escolhido para este trabalho por sua facilidade de implementação e pelos resultados positivos que se obtém com sua classificação. Gomes [2] cita que o classificador Naive Bayes é considerado um dos mais eficientes em questões relacionadas com processamento e precisão na classificação de novas amostras.

Figura 3: Teorema de Bayes

$$P(A|B) = \frac{P(B_1|A).P(B_2|A)...P(B_n|A).P(A)}{P(B_1).P(B_2)...P(B_n)}$$

A Figura 3 (acima) ilustra o teorema de Bayes. Assumindo que B representa um evento que ocorreu previamente e A um evento que depende de B, para que seja calculada a probabilidade de A ocorrer dado o evento B, o algoritmo deverá contar o número de casos em que A e B ocorrem juntos e dividir pelo número de casos em que B ocorre sozinho.

Desta forma, o algoritmo Naive Bayes será aplicado nos textos coletados e pré-processados, depois divididos em dois conjuntos, denominados *treinamento* e *teste*. A partir do conjunto de treinamento, cria-se um modelo de classificação dos *tweets* e em seguida rastreia-se as possibilidades associadas a cada sentimento, dado a presença das *hashtags* associadas a esse *tweet*. No segundo passo, usa-se o modelo criado durante o treinamento e o valida para o conjunto de teste, verificando quão bom é o modelo obtido. T*weets* que contiverem em seus textos *hashtags* ou palavras que pertencem ao dicionário pré-definido, serão analisados de acordo com a polaridade da palavra do dicionário, positiva ou negativa. Além disso, os *tweets* poderão ser classificados como ambíguos, quando eles podem ser associados a algum sentimento, mas não de maneira clara, ou neutros, quando não for possível associá-los a algum sentimento. Depois de criado e validado, o modelo está pronto para classificar novos *tweets* em uma das 4 categorias pré-definidas.

Nesta etapa, foi montado um conjunto para treino, contendo 3285 *tweets*, escolhidos randomicamente dentre todos os *tweets* colhidos ao longo dos jogos. Dois autores classificaram esses *tweets* manualmente, em uma das 4 classes definidas: positivos, quando as *hashtags* ou palavras contidas no *tweet* indicam que o texto é favorável ao evento ou resultado do jogo ou negativos caso contrário. *Tweets* que possuíam *hashtags* ou palavras

[8] http://www.agenciamestre.com/seo/stop-words-como-funcionam-palavras-de-parada
[9] https://www.princeton.edu/~achaney/tmve/wiki100k/docs/Naive_Bayes_classifier.html

que expressavam algum sentimento de uma maneira não muito clara foram classificados como ambíguos ou aqueles em que as *hashtags* ou palavras não expressavam um sentimento foram classificados como neutros.

Esse conjunto de treino, em seguida, foi submetido ao algoritmo Naive Bayes do Mahout, para gerar o modelo de classificação dos *tweets*. Após avaliação dos índices de precisão e validação do modelo, fornecidos pelo algoritmo, classificamos os demais *tweets,* e seguimos para a etapa de análise.

5.4 Análise de sentimentos

Nesta etapa, foi avaliada a classificação do sentimento expresso online da população sobre a Copa do Mundo FIFA Brasil 2014, realizada na etapa de mineração, com os fatos que ocorreram ao longo dos jogos da seleção, como resultados das partidas, manifestações, brigas de torcidas, etc.

Após a validação da classificação do modelo de treino gerado na etapa anterior, foi realizada uma análise dos fatos que ocorreram no período da coleta em relação com a classificação realizada na etapa de mineração. Aqui, validamos a classificação, com base nos fatos que ocorreram correlacionados à Copa do Mundo 2014. Por exemplo: 85% dos *tweets* coletados na partida de estréia do evento foram classificados como *tweets* com polaridade positiva. Naquele mesmo dia, a Seleção Brasileira de Futebol, time da casa, venceu a partida de estréia, causando alegria dos torcedores, que avaliaram de forma positiva a partida, em suas contas no Twitter.

6 Resultados

6.1 Coleta de dados

Durante os sete dias em que a Seleção Brasileira de Futebol disputou partidas, foram coletados 704117 *tweets* distintos. A Tabela 1 (abaixo) mostra as partidas, datas e resultados dos jogos. Para isso, foram observados *hashtags* e *trending topics* correlacionados à Copa do Mundo e aos jogos, ao longo dos dias de jogos, para que fosse possível coletar os *tweets* que falavam sobre o evento. Os *tweets* coletados foram salvos em planilhas com valores separados por tabulação. Cada planilha contém *tweets* criados em uma das datas das partidas, totalizando 7 planilhas.

Tabela 1: Data e resultados das partidas da Seleção Brasileira de Futebol

Partida	Data	Resultado
Brasil x Croácia (1º fase)	12/06/2014	3 x 1
Brasil x México (1º fase)	17/06/2014	0 x 0
Camarões x Brasil (1º fase)	23/06/2014	1 x 4
Brasil x Chile (Oitavas de Final)	28/06/2014	1 x 1 (3 x 2 pênaltis)
Brasil x Colômbia (Quartas de Final)	04/07/2014	2 x 1
Brasil x Alemanha (Semifinal)	08/07/2014	1 x 7
Brasil x Holanda (3º Lugar)	12/07/2014	0 x 3

6.2 Desenvolvimento e validação do modelo de classificação

Após o pré-processamento dos textos coletados, foram selecionados de todas as planilhas, de maneira randômica, 3285 *tweets,* para gerar o modelo de classificação. Os *tweets* desse conjunto foram classificados manualmente. Em seguida, esse conjunto de treino classificado foi dividido em duas partes, onde 80% dos *tweets* foram selecionados randomicamente para treinar o algoritmo de Naive Bayes (2634 *tweets*) e 20% dos *tweets* (651 *tweets*) foram selecionados randomicamente para testar o modelo.

Após gerado o modelo, foi testado a acurácia de classificação do mesmo. O modelo apresentou uma taxa 88,91% de precisão, utilizando o conjunto de treino contendo 2634 *tweets* para testá-lo. O mesmo

apresentou ótimas taxas de precisão para a categoria positiva (94%) de precisão, negativa (93%) e neutra (84%). Para classe ambígua, mostrou-se bom (75%).

Em contrapartida, ao ser testado com o conjunto de teste (651 *tweets*), o modelo apresentou uma taxa de 74,80% de precisão. O modelo apresentou 82% de precisão para a categoria positiva, 84% de precisão para a categoria negativa, 69% para a categoria neutra e apenas 40% de precisão para a categoria ambígua.

O modelo, assim, apresentou-se bom para classificar *tweets* positivos e negativos. No ambíguo e neutro, ocorreram uma maior taxa de erro.

Foi observado um grande número de ocorrências em que um *tweet* continha mais de uma *hashtag,* onde cada uma expressava um sentimento diferente (às vezes oposto), dificultando o trabalho de classificação para o conjunto de treino, sendo esses, uma das dificuldades encontradas para se gerar um melhor modelo de classificação.

Também fora submetida uma pequena amostra com 312 *tweets* para serem classificadas com o modelo gerado. Deste número, 167 instâncias são positivas. Foi observado que o modelo classificou corretamente 98% desta quantia (165 instâncias), mostrando-se muito eficiente para *tweets* positivos. Já, na categoria negativa, de 85 instâncias, o modelo classificou corretamente 70 (76% do total de negativas). Para a classe ambígua, de 22 instâncias, o modelo classificou corretamente 8 delas (36% do total de ambíguas), sendo esta a categoria onde a classificação foi muito ruim. Já na neutra, o modelo classificou corretamente 24 de 32 instâncias (75% do total de neutras). No geral, o modelo gerado classificou corretamente 85,57% das instâncias que compunham a amostra.

6.3 Análise de sentimento

Como mencionado, foram geradas nuvens de palavras para cada dia de jogo, no intuito representar graficamente as variações das palavras mencionadas com mais frequência ao longo dos dias de partidas, e classificado os *tweets* coletados em cada dia de partida. É possível observar na Figura 4, dia de estreia da Seleção na Copa do Mundo, uma grande alternância de palavras que expressam um sentimento positivo, como "vaibrasil" e 'rumoaohexa', com palavras negativas, fazendo alusões contra a realização do evento, como "naovaitercopa" e "fifagohome". Conferimos, na Tabela 2, que a maioria dos *tweets* do dia da estréia foram classificados como positivos. A seleção brasileira venceu a seleção croata na estreia. Entretanto, o primeiro gol da seleção brasileira foi contra, gerando um furor entre os usuários no Twitter. Podemos observar, a palavra "gol contra" aparece entre as mais frequentes.

Figura 4: Palavras mais frequentes do dia 12/06/2014 – Brasil x Croácia

Tabela 2: Classificação dos *tweets* – 12/06/2014

Classe	Quantidade
Positivo	37920 (64,22%)

Negativo	11842 (20,05%)
Ambíguo	2729 (4,63%)
Neutro	6554 (11,10%)
Total	59045 (100%)

As palavras de apoio seguem até o último dia de partida da seleção nacional, como é possível observar nas Figuras 5,6,7,8,9 e 10, e nas tabelas 3,4,5,6,7 e 8, enquanto as que fazem alusão contra a Copa vão se destacando menos. Podemos conferir também, consideráveis números de *tweets* classificados como neutros, e uma pequena quantidade classificada como ambíguo ao longo das partidas.

Figura 5: Palavras mais frequentes - 17/06/2014 – BRA x MEX

Tabela 3: Classificação dos tweets – 17/06/2014

Classe	Quantidade
Positivo	37985 (39,55%)
Negativo	7505 (7,81%)
Ambíguo	13755 (14,32%)
Neutro	36781 (38,30%)
Total	96026 (100%)

Podemos ver que em todas as nuvens de palavra aparecem menções ao jogador Neymar Júnior (como nas Figuras 6,7 e 8). O mesmo é considerado um dos melhores jogadores da seleção brasileira. No dia do jogo contra a seleção colombiana, o mesmo sofreu uma lesão após uma falta grave[10]. Podemos observar as menções "neymarjr" e "neymar" entre os assuntos de maior destaque na nuvem de palavras referentes ao dia daquele jogo (mais abaixo, na Figura 8), mostrando a grande popularidade do mesmo.

Figura 6: Palavras mais frequentes: 23/06/2014 – CAM x BRA

[10] http://copadomundo.uol.com.br/noticias/redacao/2014/07/04/neymar-esta-fora-da-copa-do-mundo-apos-fraturar-vertebra.htm

Tabela 4: Classificação dos *tweets* – 23/06/2014

Classe	Quantidade
Positivo	46703 (34,67%)
Negativo	7676 (5,69%)
Ambíguo	7483 (5,55%)
Neutro	72810 (54,06%)
Total	134672 (100%)

Figura 7: Palavras mais frequentes - 28/06/2014 – BRA x CHI

Tabela 5: Classificação dos *tweets* – 28/06/2014

Classe	Quantidade
Positivo	48944 (38,99%)
Negativo	8635 (6,87%)
Ambíguo	10492 (8,35%)
Neutro	57446 (45,76%)
Total	125517 (100%)

Figura 8: Palavras mais frequentes - 04/07/2014 – BRA x COL

Tabela 6: Classificação dos *tweets* – 04/07/2014

Classe	Quantidade
Positivo	64925 (46,23%)
Negativo	7165 (5,10 %)
Ambíguo	9679 (6,89%)
Neutro	58652 (41,76%)
Total	140421 (100%)

No dia 08/07/2014, palavras com teor negativo como ’vergonha’ e ‘vergonhabrasil’ ganham destaque (Figura 9). O número de *tweets* classificados como negativos sobe consideravelmente em relação aos dias de partida anteriores, como podemos conferir na Tabela 7. Na ocasião, a seleção brasileira perdeu a sua primeira partida já na semifinal do evento, contra a seleção alemã por 7x1, maior derrota já registrada na história do time[11]. Em mesma proporção, palavras de apoio ao jogador da seleção brasileira David Luiz, considerado destaque do time no evento, ganharam os assuntos do momento no Twitter, após a derrota da seleção. Alguns noticiários o trataram como o jogador mais querido do Brasil[12].

Parte da população brasileira acreditava que a Copa havia sido comprada pela Presidente do Brasil Dilma Russeff, isso se comprova por uma *hashtag* frequente que ocorreu quando a seleção brasileira perdeu para a seleção da Alemanha por 7 x 1 Observe essa *hashtag* na Figura 9 ‘dilmaesqueceudepagaralemanha’. É possível perceber como o conteúdo das redes sociais refletiu o sentimento e o pensamento da população brasileira durante o evento.

Figura 9: Palavras mais frequentes - 08/07/2014 – BRA x ALE

[11] http://placar.abril.com.br/materia/derrota-para-a-alemanha-foi-a-10a-maior-goleada-em-copas
[12] http://trivela.uol.com.br/david-luiz-o-jogador-mais-amado-brasil-em-6-imagens/

Tabela 7: Classificação dos *tweets* – 08/07/2014

Classe	Quantidade
Positivo	45701 (45,38%)
Negativo	20864 (20,71%)
Ambíguo	12884 (12,79%)
Neutro	21250 (21,10%)
Total	100699 (100%)

No último dia de partida da seleção brasileira, disputando o 3º lugar do mundial contra a seleção holandesa, palavras de apoio ao jogador David Luiz continuam em alta no Twitter. Palavras negativas como 'vergonha' continuaram entre as mais mencionadas. A seleção, neste dia, perdeu novamente, ficando com o 4º lugar da Copa do Mundo[13]. Observe que na Figura 9, palavras como 'presosdacopa' e 'liberdade' ficaram em alta no último dia de participação da seleção brasileira na Copa. Nesse mesmo dia, policiais civis da cidade do Rio de Janeiro realizaram a prisão de 19 manifestantes, suspeitos de atos de vandalismos durante manifestações desde junho de 2013, ano em que ocorreu a Copa das Confederações, evento futebolístico promovido pela FIFA, no país. Vários usuários do Twitter, manifestaram-se contra as prisões dos suspeitos presos.

Figura 10: Palavras mais frequentes - 12/07/2014 – BRA x HOL

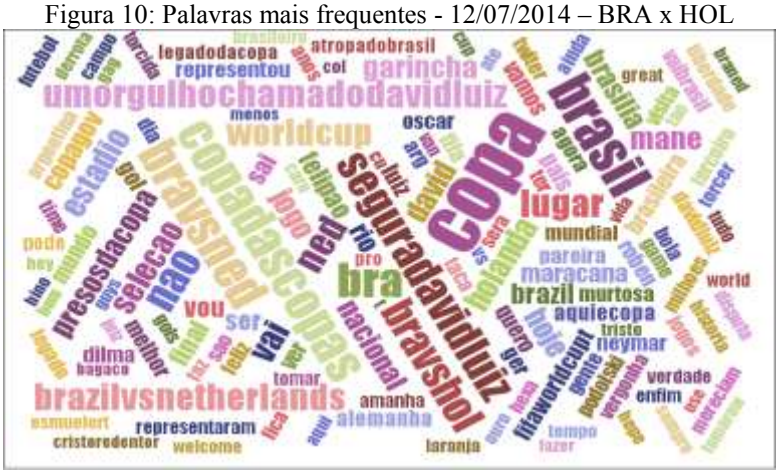


Tabela 8: Classificação dos *tweets* – 12/07/2014

Classe	Quantidade
Positivo	14553 (30,46%)
Negativo	5914 (12,38%)
Ambíguo	7716 (16,15%)
Neutro	19584 (40,99%)
Total	47767 (100%)

7 Conclusão

Este trabalho apresentou como o processo de Mineração de Textos foi usado para coletar, estruturar o texto extraído do Twitter (tweets) e como criar um modelo de classificação de texto que permita predizer a opinião da rede social do usuário do Twitter sobre Copa do Mundo da FIFA Brasil 2014. As postagens dos usuários, popularmente conhecido como tweets, são categorizadas neste trabalho como um sentimento: positivo, negativo, ambíguas ou neutras. É possível aperfeiçoar o modelo de classificação, retroalimentando o conjunto de treino

[13] http://www.jj.com.br/noticias-3091-vaiada-selecao-brasileira-perde-para-a-holanda-por-3-a-0

com mais *tweets*, balanceando o número de *tweets* para cada classe. Outras categorias também podem ser definidas, de acordo com o contexto dos dados, utilizando o algoritmo Naive Bayes.

As nuvens de palavras referentes às principais *hashtags* e palavras mais utilizadas no Twitter durante os dias de jogo da seleção brasileira e os números das classificações dos *tweets* ao longo dos dias das partidas também foram apresentadas nesse trabalho.

O modelo de classificação gerado neste trabalho nos permitiu mostrar a opinião dos usuários ao longo das partidas da seleção brasileira, validando as classificações feitas pelo modelo com os fatos associados a cada dia de partida, como o fato da seleção ter sido desclassificada na semifinal contra a seleção alemã, após perder de goleada, influenciou no aumento do número de *tweets* negativos, que até então eram inexpressivos.

Foi possível também, através das nuvens de palavras geradas, explicar a ocorrência de *hashtags* e palavras positivas e negativas, com acontecimentos relacionados à Copa, como o fato da *hashtag* ‘ forcaneymar’ ter sido mais frequente em um determinado dia, estar ligado a uma contusão que forçou o jogador Neymar Júnior a sair da competição.

Assim, o processo apresentado neste trabalho, pode ser seguido por organizações para mapear a opinião de usuários do Twitter, fazendo o uso dos resultados para os mais diversos fins dentro das mesmas.

Referências

[1] RUSSEL, Mathew A. *Mining the social web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub and More*. 2 ed. Sebastopol: O'reilly Media, Inc., 2013

[2] TAN, Ah-Hwee. Text mining: The state of the art and the challenges. PROCEEDINGS OF THE PAKDD 1999 WORKSHOP ON KWNOLEDGE DISCOVERY FROM ADVANCED DATABASES, Beijing. 1999.

[3] GOMES, Helder Joaquim Carvalheira. Text Mining: análise de sentimentos na classificação de notícias. *Information Systems and Technologies (CISTI),* 2013 8th Iberian Conference on. Lisboa. 2013.

[4] HEARST, M. A. Untangling text data mining. PROCEEDINGS OF THE 37th ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS ON COMPUTATIONAL LINGUISTICS (pp. 3–10),1999. Stroudsburg, PA, USA: Association for Computational Linguistics.

[5] LIDDY, E. *Natural Language Processing. Encyclopedia of Library and Information Science*. New York: Marcel Decker, Inc, 2001.

[6] MORAIS, Edilson Andrade Martins; AMBRÓSIO, Ana Paula L. *Mineração de Textos*. Goiânia: UFG. 2007. (Série Texto Técnico, INF_005/07).

[7] INDURKHYA, Nitin; DAMERAU, Fred J. *Handbook of natural language processing.* 2ed. Florida: CRC Press, 2010.666 p.

[8] BARBOSA, Glívia Angélica Rodrigues et al. Characterizing the effectiveness of twitter hashtags to detect and track online population sentiment. In: PROCEEDINGS OF THE 2012 ACM ANNUAL CONFERENCE EXTENDED ABSTRACTS ON HUMAN FACTORS IN COMPUTING SYSTEMS EXTENDED ABSTRACTS**.** Austin, 2012.

[9] ARANHA, C.N. *Uma abordagem de Pré-Processamento Automático para Mineração de Textos em Português: Sob o Enfoque da Inteligência Computacional*. 2007. 144 f. Tese (Doutorado em Engenharia Elétrica) – Pontífica Universidade Católica do Rio de Janeiro, Rio de Janeiro. 2007.

[10] OWEN, Sean et al. *Mahout in Action*. Connecticut: Manning Publications Co, 2011. 373p.

Interface de Aplicação NFC em Hardware Livre para Pagamentos Móveis

Carlos David Braga Borges [1]
Jair Alves de Carvalho [1]
Arthur Sousa de Sena [1]
Prof. Dr. José Cláudio do Nascimento [2]

Resumo: Neste trabalho nós apresentamos resultados de testes de comunicação sob uma plataforma de hardware livre (Arduino UNO e Mega com shield NFC) para verificar a potencialidade de aplicação desse hardware na construção de interfaces para sistemas de pagamento móveis. A importância do hardware livre em um sistema de segurança tem suas raízes no princípio de Kerckhoffs que diz que o sigilo do design não é uma garantia de segurança. Verificamos que o Arduino pode ser aplicado em projetos de protocolos seguros, porém protocolos mais robustos podem exigir maior capacidade de canal. O Arduino com shield NFC testado está limitado a transmitir apenas 120 bytes e numa transmissão multipacotes atinge uma taxa média de transmissão de 110.75 Bps. Para protocolos mais robustos, um módulo Bluetooth pode ser adicionado, no entanto a espionagem do canal é facilitada (exigindo algoritmos de cifras).

***Palavras-chave*:** Hardware Livre. NFC. Pagamentos Móveis.

Abstract: *In this work we present the results of communication tests under a free hardware platform (Arduino UNO and MEGA with NFC shield) to verify the potential of its application in the building of mobile payment systems. The importance of free hardware in a security system has its roots in the Kerckhoffs principle, which states that secrecy of design is not a guarantee of security. We verify that Arduino can be applied in projects of secure protocols, but more robust protocols may require greater channel capacity. The Arduino with NFC shield we tested is limited to transmit only 120 bytes in a single transmission and in a multipacket transmission it has an average transmission rate of 110.74 Bps. For more robust protocols, a Bluetooth module may be added, however, in this case, channel eavesdropping is easier (requiring cipher algorithms).*

***Keywords*:** *Free Hardware. Mobile Payments. NFC.*

1 Introdução

Comunicação de campo próximo, popularmente conhecida como NFC, abreviação para "Near Field Communication", é um conjunto de padrões de comunicação via rádio ativado quando dois dispositivos estão bem próximos. O padrão NFC suporta taxa de dados de 106Kbps, 212 Kbps e 424 Kbps, operando em curtas distâncias na frequência de 13.56MHz.

A principal motivação para a implementação dessa tecnologia foi o comércio eletrônico. A promessa é substituir os cartões de crédito no futuro próximo. Assim, muitos cartões podem estar presentes num só aplicativo, tornando mais prática qualquer operação de pagamento. Mas outras aplicações foram surgindo ao longo do desenvolvimento. E a segurança tem sido rigorosamente discutida.

[1]Curso de Engenharia de Computação (CEC), UFC, Campus Mucambinho - Rua Coronel Estanislau Frota s/n - Sobral (CE) - Brasil
[2]Curso de Engenharia Elétrica (CEE), UFC, Campus Mucambinho - Rua Coronel Estanislau Frota s/n - Sobral (CE) - Brasil {claudio.nasce@gmail.com}

Em 1883 Auguste Kerckhoffs escreveu seis princípios para projetos de sistemas de cifras [1]. Mas um deles pode ser generalizado para muitos algoritmos de segurança no qual ele afirmou: "É necessário que o sistema em si não requeira sigilo, e que o mesmo possa cair sem desvantagem em mãos inimigas." Essa condição não foi atendida por muitas empresas no final do século XX, quando muitos programadores usando engenharia social conseguiram os códigos do fabricante para desenvolver muitas técnicas de ataques a sistemas de comunicações. Mas esse não foi o caminho tomado no início do desenvolvimento do NFC. Em 2004 foi criado o NFC Forum. Onde atualmente todas as especificações do sistema estão abertas para discussões de implementação de qualquer sistema antes dele entrar no mercado. Os 190 associados em conjunto discutem a implementação dessa tecnologia de forma aberta com desenvolvedores e futuros usuários.

Por outro lado, na busca competitiva para por no mercado o primeiro sistema de pagamento móvel, outros caminhos para a implementação têm sido seguidos. O Google anunciou, apenas em 2013 a sua participação no NFC Forum [2], mas em 2011 já havia lançado independentemente o Google Wallet [3]. Um dia após o lançamento, foi acusado pelo PayPal e eBay de roubo de projetos por aliciamento de funcionários. Percebemos nessa disputa que muitas destas características sigilosas são detalhes de implementação do protocolo. Eles são escondidos apenas para evitar a engenharia reversa do concorrente.

Ainda hoje, embora o Google tenha entrado no NFC Forum, muitas características do desenvolvimento NFC para Android não foram abertas para desenvolvedores [4]. As APIs para NFC estão limitadas a aplicações simples (leituras de tags) e não permitem aplicações mais elaboradas como sistemas de pagamento. Isso faz com que a afirmativa de segurança do sistema completo seja uma incógnita. A prova disso é que falhas de segurança foram notadas um ano depois do lançamento e anunciadas na internet [5], sem contar as que não são anunciadas para a vantagem dos que se beneficiam delas.

Devido ao interesse financeiro, a implementação de sistemas de pagamento móveis está seguindo uma história bem diferente do PGP. Cujo código aberto (OpenPGP) e a confiabilidade vinda do fracasso de ataques promovidos por instituições renomadas na área de segurança, criaram uma credibilidade para o público de internet que deseja sigilo em suas comunicações de email. Assim, como no projeto PGP, os sistemas de pagamento móveis devem entrar na categoria de software e hardware livre. Quando o sigilo de dados é necessário, é do interesse dos usuários saber que o sistema usado é realmente seguro. Protocolos comprovadamente seguros mesmo quando seus códigos são abertos é uma forte garantia de segurança para o usuário.

O RC-S380 da Sony é o primeiro leitor certificado pelo Programa de Certificação NFC Forum. Embora não seja de hardware livre, ele foi projetado para ser compatível com diversos dispositivos. Neste trabalho, nós testamos uma interface NFC de leitura e escrita sob Arduino (hardware livre). Também implementamos a interface com Bluetooth. O desenvolvimento sob uma plataforma de hardware livre (Arduino Uno e shield NFC e módulo Bluetooth) toda a liberdade para testes de segurança. Até agora, pelo nosso conhecimento, essa plataforma ainda não foi testada para fins de pagamento móvel. Devido a ser um hardware proposto para aplicações mais simples, nós testamos a capacidade de canal das duas interfaces e testamos as suas limitações. Verificamos que seria possível uma implementação de um sistema seguro com os mesmos níveis de segurança aos usados na internet, onde assinaturas digitais, códigos de autenticação, textos cifrados e certificados digitais são transmitidos.

2 Modelo de segurança para a interface de pagamento NFC na plataforma Arduino

Em um sistema de pagamento digital temos três entidades: usuário, vendedor e corretor. Para a análise de segurança, deve-se imaginar vários cenários em que cada entidade pode assumir atitudes desonestas. Não entrare-

mos nos detalhes desses ataques. Mas durante a análise de caso, basicamente três exigências devem ser satisfeitas: sigilo, autenticidade e integridade. Seja na geração, transmissão ou no armazenamento dos dados. Dependendo do sistema, algumas exigências são mais essenciais do que outras, mas elas sempre devem ser suficientemente satisfeitas. No projeto de comunicação NFC P2P para pagamento digitais, devemos levar em consideração as características do canal, da fonte e do receptor.

Quanto ao módulo Arduino, no projeto jamais devemos considerá-lo como fonte ou receptor seguro de dados. Assim, devemos evitar que ele gere ou armazene dados que não podem ser publicamente revelados. Ele não se encaixa no modelo de hardware seguro, o que nos leva a assumir que memória, barramento e cabos são acessíveis a qualquer intruso no sistema.

Quanto ao canal de comunicação, o sinal de rádio frequência para transferência de dados pode ser captado. A distância é um parâmetro que define se o espião do canal está hábil ou não para escutar o canal [6]. Pois a potência do sinal que chega no receptor é importante nesse ataque. Ela define a qualidade da informação. Para a comunicação NFC uma antena pode ser projetada para um ataque sem contato com qualquer dispositivo dentro de uma distância de 10m [7]. Apenas um pouco de engenharia social é necessária para esconder a antena. Nessas condições, não pode-se afirmar que o canal NFC oferece sigilo. Embora a escuta do canal não seja uma tarefa simples, ela não é impossível. Portanto, a cifra de mensagens ainda será uma um recurso necessário para garantir o sigilo da comunicação nesse canal. O recurso comumente utilizado para resolver esse tipo de problema é a criptografia simétrica, usando algoritmos como AES, Blowfish, Twofish, entre outros.

Também, em NFC, é fácil destruir dados. É impossível evitar esses ataques, mas é possível identificá-los. Assim, técnicas para garantia de integridade serão necessárias. Isso envolve o uso de funções de hash, como SHA-2, SHA-3 ou Whirlpool. No entanto, considerando que os recursos de um canal NFC, normalmente, são escassos, o código de saída da função de hash precisa ter um comprimento reduzido e ainda assim fornecer um bom nível de segurança.

Quanto a escolha de algoritmos para a autenticidade sabemos que uma infraestrutura para criptografia de chave pública é montada com a presença de uma autoridade confiável. O desafio nessa área para a comunicação NFC refere-se à capacidade do canal. Dentre os dados que precisam ser trocados durante protocolos de chave pública, os certificados digitais possuem o maior volume, exigindo maior capacidade do canal para transmiti-los. Em [8] são propostos certificados digitais compactos para as comunicações móveis. Exemplo é o X.509 compacto que está limitado a 88 bytes.

Dentro das condições discutidas acima, para fins de projeto, devemos tratar a interface de módulo Arduino juntamente com o canal de rádio frequência de campo próximo como um canal inseguro. Ou seja, no protocolo deve-se implementar algoritmos de segurança na máquina do vendedor e no smartphone do usuário, mas não na interface proposta nesse artigo.

3 Arquitetura da interface

O dispositivo móvel usado durante os testes foi Sony Xperia L, modelo C2104, executando Android 4.2.2. Como interface, um shield NFC com um módulo transceiver PN532 integrado, capaz de comunicação sem contato na frequência 13.56 MHz. O shield foi acoplado ao Arduino Uno (Microcontrolador ATmega328, 32 KB de memória flash, 2 KB de SRAM, 1 KB de EEPROM, taxa de clock de 16 MHz).

O shield NFC conecta-se ao Arduino através da interface SPI (Serial Peripheral Interface). O Arduino Uno

controla os dados recebidos e transmitidos pela interface NFC através da antena. O shield suporta os protocolos ISO14443 Tipo A e Tipo B, agindo como um cartão de escrita/leitura, cartão de proximidade (PICC) e suporta comunicação Peer to Peer (P2P).

Um programa escrito na linguagem do Arduino, muito similar a C/C++ é carregado no microcontrolador ATmega, acoplado ao Arduino. O programa controla a escrita e leitura de mensagens no padrão NFC Data Exchange Format (NDEF), que serão transmitidas durante o processo de compra. Esse programa aplica o Simple NDEF Exchange Protocol (SNEP) para estabelecer a comunicação e executa comandos que invocam as interfaces providas pelo firmware do PN532, através da SPI. Todos esses comandos estão documentados em [9]. Dois comandos principais foram utilizados durante os testes e serão descritos a seguir.

O comando TgSetData é usado para escrever informação na antena PCB, que será posteriormente enviada ao smartphone com NFC ativado. Neste passo, smartphone também funciona como um cartão de escrita/leitura ISO14443, o que permite comunicação P2P entre o dispositivo móvel e a interface. O segundo comando é TgGetData, que é usado para ler a informação vinda do smartphone.

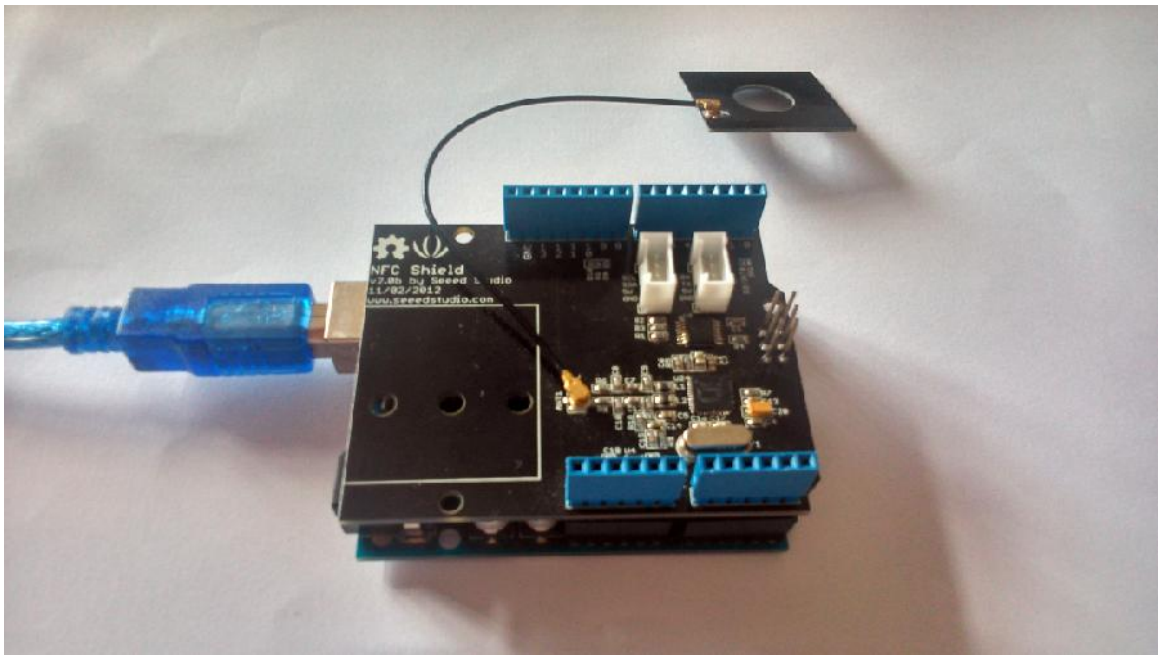


Figura 1: Shield NFC conectado à placa Arduino Uno

Em nossa proposta inicial, o Arduino é conectado ao computador do vendedor através de uma conexão USB. O software instalado na máquina do vendedor é capaz de comunicar-se com a interface através da porta especificada. Para iniciar uma transação, o vendedor digita o preço do produto utilizando o software. A seguir, durante a execução do protocolo de transação, quando o programa do vendedor necessita enviar uma mensagem encriptada à aplicação no dispositivo móvel, primeiramente a mensagem é enviada à interface através da porta USB. A interface, então, escreve os dados no cartão NFC do módulo NP532 e é transmitida ao smartphone do cliente quando este entrar na zona de proximidade da interface. Quando a aplicação do cliente precisa enviar uma mensagem encriptada ao vendedor, essa mensagem é escrita no cartão NFC do smartphone e transmitida à interface quando o smartphone é aproximado. As mensagens enviadas pela aplicação do cliente são recebidas pelo shield

NFC, entregues ao Arduino e automaticamente transmitidas ao software do vendedor

Os processos de encriptação e decriptação são realizados apenas no software do vendedor e na aplicação do dispositivo móvel do cliente.

4 Capacidade do canal NFC

Com o propósito de simular as transmissões entre um vendedor e um usuário, um servidor, desenvolvido em Java e instalado no computador desktop do vendedor, envia dados a um dispositivo Android através da interface Arduino Uno e shield NFC. Devido a uma limitação do módulo NFC que utilizamos, apenas é possível receber e enviar 256 bytes de cada vez, onde 136 bytes são ocupados pelo cabeçalho NDEF padrão. Então, efetivamente, é possível enviar, em apenas uma escrita, no máximo 120 bytes de informação referente à transação. Portanto, se a informação a ser transmitida excede 120 bytes, ela deve ser dividida em mútiplos pacotes de 120 bytes e transmitida através de escritas múltiplas. Para testar a velocidade e eficiência desse sistema, realizamos uma grande quantidade de experimentos medindo o intervalo de tempo entre o envio e recebimento de pacotes com tamanhos diferentes, do shield NFC para o dispositivo Android.

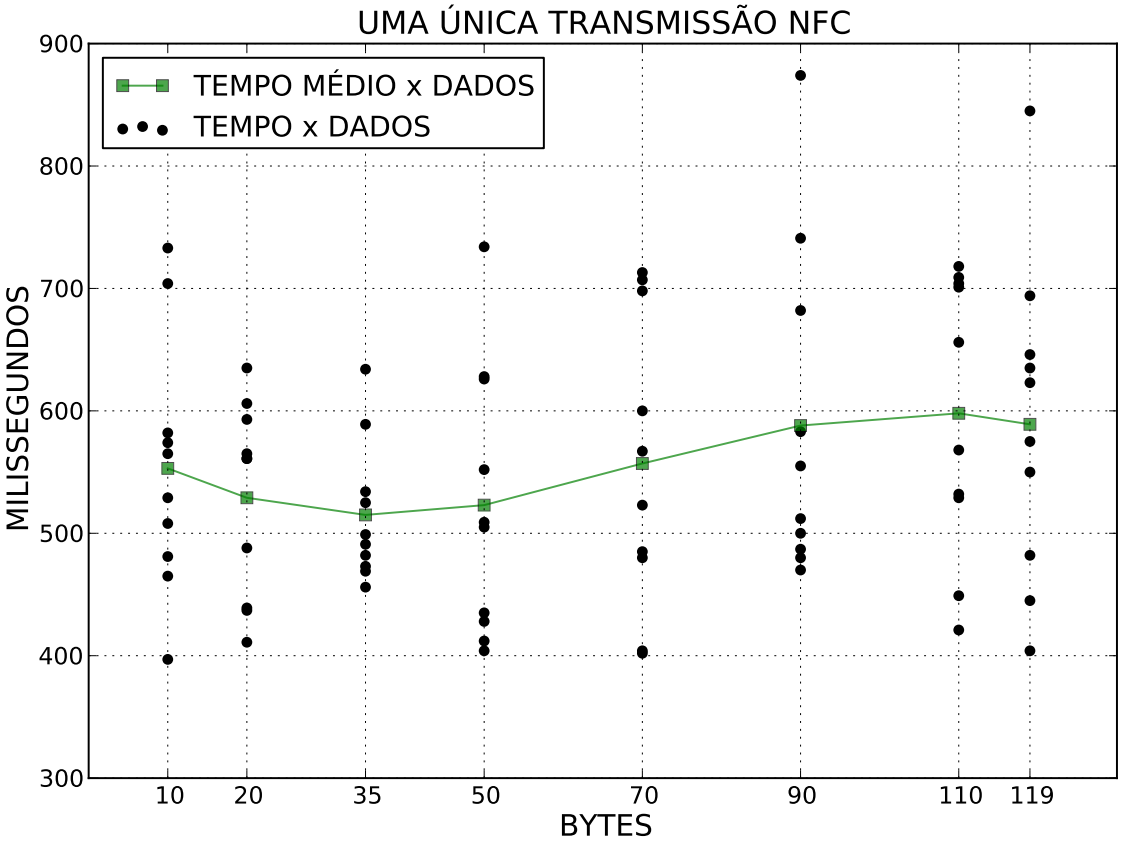


Figura 2: Resultados experimentais para apenas uma escrita

A proximidade entre o dispositivo móvel e a interface Arduino depende do usuário, então a distância é uma variável aleatória. Já que a comunicação NFC depende da proximidade, temos uma variação da capacidade do canal de acordo com a distância. Por exemplo, em uma simulação em [10], a capacidade do canal é cerca de 100

Kbps a uma distância de 5 cm e decresce para menos de 10 Kbps a uma distância de 20 cm. Os resultados obtidos em [10] referem-se à taxa de dados do NFC. Então, executamos experimentos para verificar o tempo e taxa médios da interface Arduino e shield NFC como um todo. Isso significa que os tempos obtidos aqui medem delays na transmissão serial do Arduino, a escrita no cartão NFC, além da própria transmissão via NFC.

Para realizar os experimentos, o smartphone foi colocado acima da antena do shield NFC, mantendo os cartões NFC próximos. As medidas de tempo forma feitas usando a função *millis()* no código do Arduino, com a transmissão sendo realizada do Arduino para o dispositivo Android. Essas medidas estimam o delay na SPI, o tempo de escrita e o estabelecimento da conexão. Os resultados foram mostrados no computador através do Monitor Serial do Arduino. Então, com o tempo calculado no Arduino, a taxa de transmissão da interface pode ser computada.

Na figura 2, expressamos por pontos pretos os resultados obtidos com o envio de pacotes de até 120 bytes de informação efetiva, realizados com apenas uma escrita do módulo NFC. Em verde, temos a taxa de transmissão média para cada pacote. Observando a figura 2, percebe-se que independentemente do tamanho do pacote, a transmissão ocorre dentro de um intervalo de tempo entre 400 e 900 milissegundos. O tempo médio obtido em apenas uma escrita foi 557.1 milissegundos.

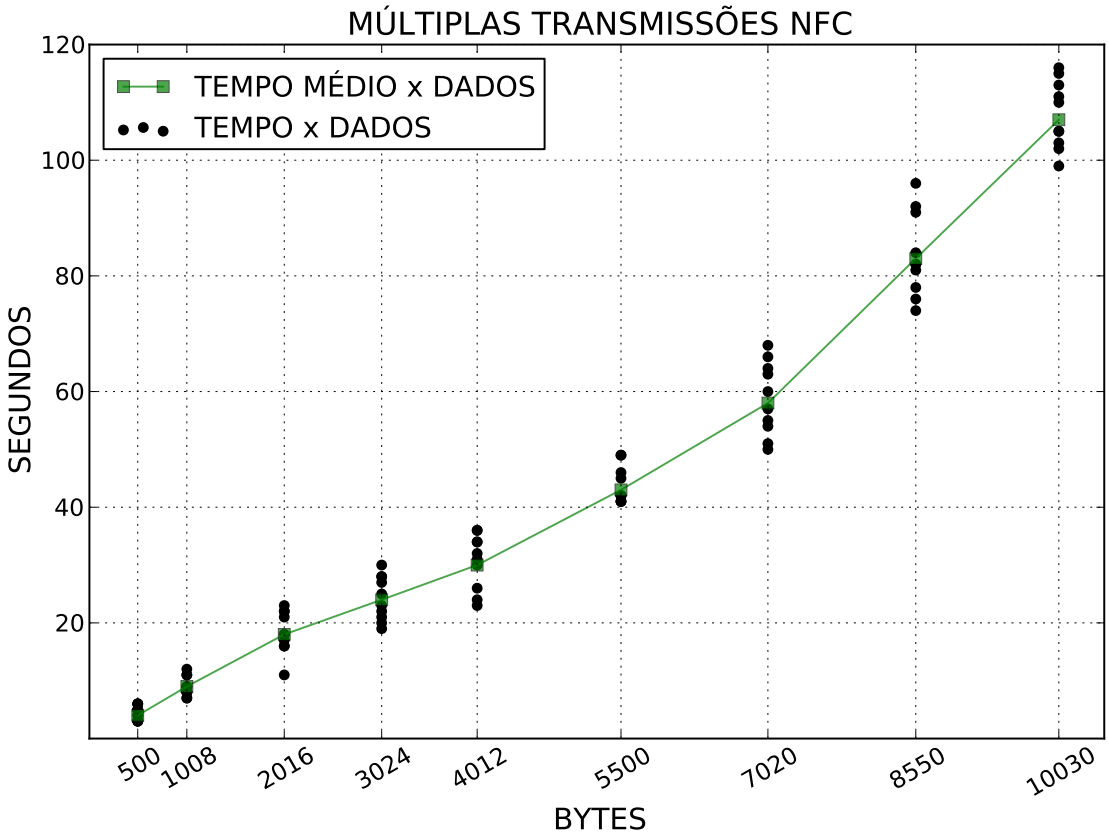


Figura 3: Resultados experimentais para escritas múltiplas

A figura 3 mostra os resultados obtidos para dados maiores que 120 bytes, em que a transmissão ocorre através de escritas múltiplas. É perceptível que o tempo médio de transmissão aumentar quase linearmente com relação ao tamanho da informação. Quanto maior a quantidade de informação, mais escritas são necessárias para

finalizar a transferência. Em adição, a cada transmissão, uma nova sequência de handshakes entre o shield NFC e o smartphone Android deve ser realizada. Esse processo consome bastante tempo se tiver de realizado múltiplas vezes. Além disso, o tempo requerido para escrita dos dados no módulo PN532 é aumentado pelo delay da interface SPI. Essas razões são as principais explicações para a baixa taxa de transmissão efetiva de 110.75 B/s da interface Arduino Uno e shield NFC.

5 A solução Bluetooth

Os resultados obtidos na seção anterior apontam que usar apenas Arduino e shield NFC com módulo PN532 como interface não permite a execução rápida de um protocolo de segurança que exija transmissão de dados superior a 120 bytes. Uma solução pode ser montada através de adição da funcionalidade Bluetooth à interface. Assim, um módulo Bluetooth seria conectado ao Arduino e usado para executar os trechos do protocolo que necessitam de transmissões de dados maiores que 120 bytes. A comunicação NFC, nesse caso, poderia ser utilizada como uma forma de iniciar automaticamente uma conexão Bluetooth entre o módulo da interface e o dispositivo móvel. Por exemplo, a aproximação entre o smartphone e a interface pode servir para compartilhar o endereço MAC do hardware Bluetooth de um dos dispositivos, permitindo assim a conexão entre os dois aparelhos. Considerando que o endereço MAC de qualquer hardware Bluetooth pode ser encapsulado em 6 bytes, sua transmissão utilizando o canal NFC provido pelo shield com módulo PN532 pode ser realizada em apenas uma escrita.

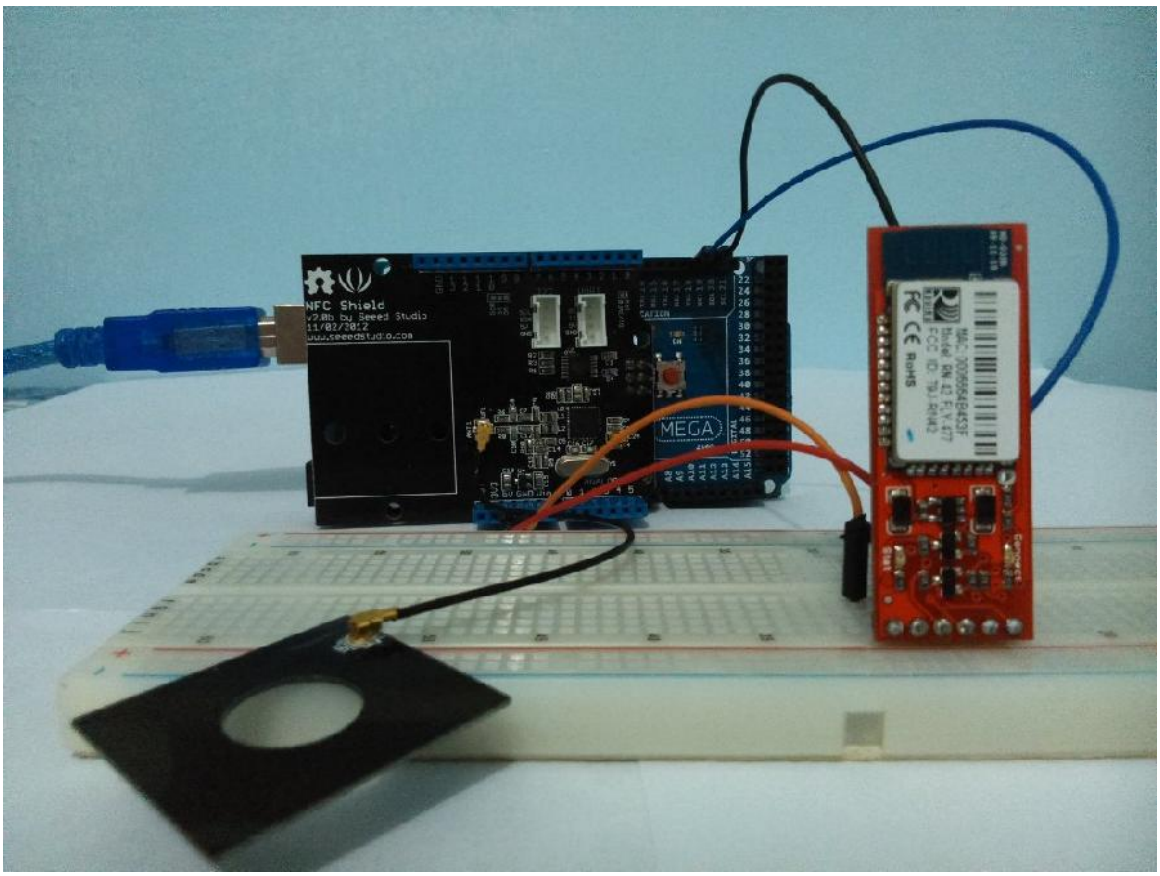


Figura 4: Shield NFC e módulo Bluetooth conectados a um Arduino MEGA.

Nesta segunda proposta, usamos um Arduino MEGA (Microcontrolador ATmega2560, 256 KB de memória flash, 8 KB de SRAM, 4 KB de EEPROM, taxa de clock de 16 MHz), já que este possui mais pinos RX/TX para comunicação serial do que o Arduino Uno. Essa medida visa a evitar interferências entre os canais de comunicação serial. Usamos o módulo Bluetooth Mate Silver da Sparkfun, que utiliza o hardware Bluetooth RN-42, para testar a capacidade de transmissão dessa interface.

O módulo Bluetooth foi alimentado pelo pino 5V do Arduino e conectado pelos pinos RX1 e TX1 para comunicação serial. Assim, o Bluetooth exerce a função que o shield NFC exercia na proposta anterior, intermediar o trânsito de dados entre o software instalado na máquina do vendedor e a aplicação no dispositivo módel do cliente.

No experimento com Bluetooth, as medidas de tempo foram feitas usando a função *micros()* no código do Arduino, com a transmissão sendo realizada do Arduino para o dispositivo Android. As medidas estimam os delays da comunicação serial, além do tempo da transmissão Bluetooth. Os resultados obtidos puderam ser visualizados através do Monitor Serial do Arduino.

A figura 5 exibe os resultados do experimento com o módulo Bluetooth. Os pontos pretos são de difícil visualização, pois os resultados são muitos próximos da média, ou seja, não apresentam uma variância considerável. Os experimentos foram realizados com uma única escrita e nota-se a taxa de transmissão constante através da tendência linear do gráfico. A taxa de dados efetiva obtida na solução com módulo Bluetooth é aproximadamente 11400 B/s, bastante superior à taxa obtida com o shield NFC.

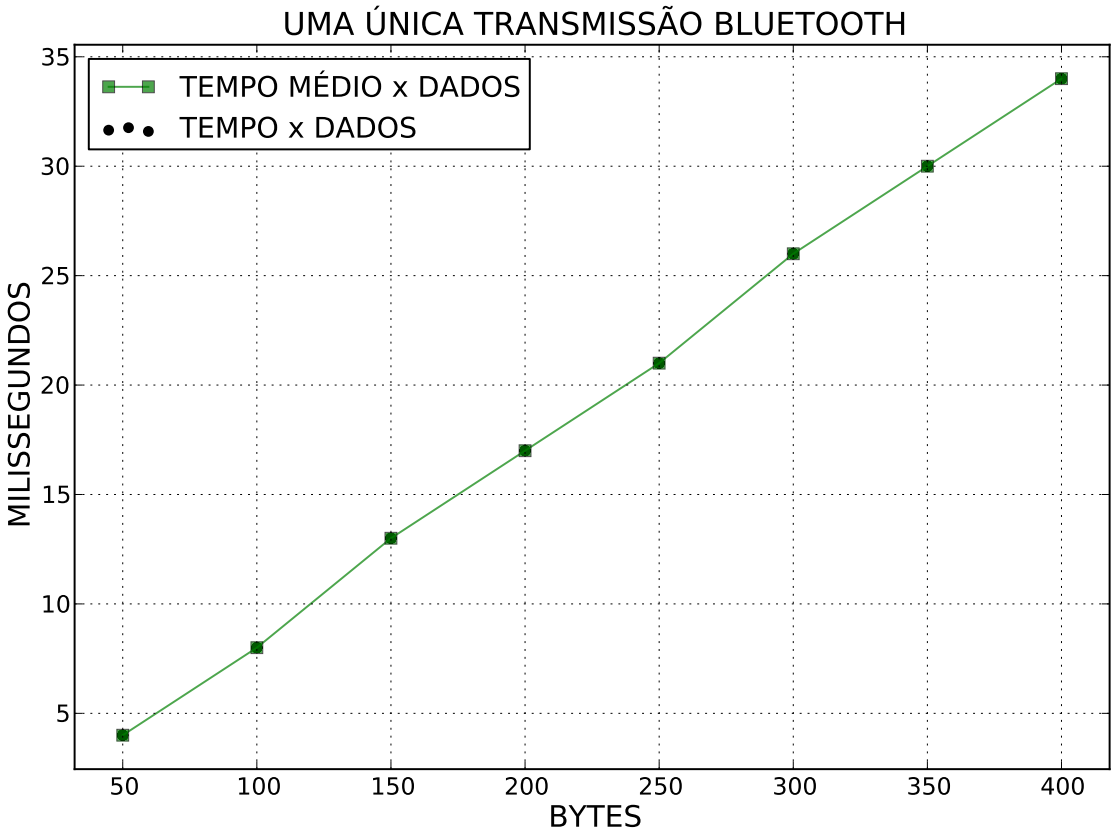


Figura 5: Resultados experimentais para uma única transmissão Bluetooth

6 Conclusão

Neste trabalho nós apresentamos testes sob uma plataforma de hardware livre para verificar a potencialidade de aplicação a sistemas de pagamento móveis. Em sistemas de segurança da informação é essencial que os algoritmos sejam abertos e ainda seguros nessas condições, pois o sigilo do design não é uma garantia de segurança.

Ao fazer uma transmissão, o Arduino com shield NFC testado está limitado a transmitir apenas 120 bytes. Para transmissões maiores é necessário dividir os dados em pacotes de 120 bytes. Numa transmissão de apenas um pacote de diferentes tamanhos, porém menores que 120 bytes, observamos que o tempo varia entre 400ms e 900ms. O tempo médio obtido foi 557.1 ms, mas a taxa de transmissão pode chegar até 300B/s. Já quando a transmissão é feita em multipacotes a taxa média chega a 110.75 Bps.

Verificamos que o Arduino pode ser aplicado em projetos de protocolos seguros, porém protocolos mais robustos exigirão maior capacidade de canal. A utilização de algoritmos de criptografia simétrica, assimétrica e hashes criptográficos com nível de segurança semelhantes ao que temos atualmente na Internet exige a transmissão e recepção de alguns quilobytes de informação, onde o maior volume de dados concentra-se, geralmente, nos certificados. Isso implica que um sistema de maior segurança e melhor performance exige que o hardware seja capaz de lidar com taxas de alguns quilobytes por segundo. Ficando como perspectiva futura o desenvolvimento de novos shields que possam suportar taxas de transmissão de dados mais próximas do padrão NFC. Pode-se alcançar taxas de transmissão de dados de 106 kbps, 212 kbps e 424 kbps. Há a necessidade de desenvolver um hardware livre de melhor performance para aplicação em pagamentos móveis.

O acréscimo do módulo Bluetooth aumenta a capacidade do canal de comunicação. A taxa de dados efetiva obtida na solução com módulo Bluetooth é aproximadamente 11400 B/s, bastante superior à taxa obtida com o shield NFC. Mas isso favorece a escuta do canal. Embora, a escuta do canal NFC seja possível ela não é tão simples. Exige a construção de antenas apropriadas e uma engeharia social trabalhosa para esconder a antena. Enquanto que o Bluetooth pode ser facilmente escutado. Mas a escuta do canal Bluetooth não será um problema se ela se tornar sigilosa através de cifras.

7 Agradecimentos

Este trabalho foi apoiado pela Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP) e o Programa de Educação Tutorial da Engenharia da Computação da Universidade Federal do Ceará (PET-UFC).

Referências

1 KERCKHOFFS, A. La cryptographie militaire. *Journal des sciences militaires*, IX, p. 161–191, 1883.

2 CASSIDY, R. Nfc forum welcomes google to board of directors. In: . [S.l.: s.n.], 2013.

3 CIRIACO, D. Paypal e ebay acusam google de roubo de segredo comercial. In: . [s.n.], 2011. Disponível em: <http://www.tecmundo.com.br/google/10336-paypal-e-ebay-acusam-google-de-roubo-de-segredos-comerciais.htm>.

4 WOODY. Enable real nfc p2p communication and the option to disable the "touch to beam" ui. In: . [s.n.], 2012. Disponível em: <https://code.google.com/p/android/issues/detail?id=28014>.

5 JR., J. P. M. Cracking open google wallet. In: . [s.n.], 2012. Disponível em: <http://www.technewsworld.com/story/74408.html>.

6 HANCKE, G. et al. Eavesdropping attacks on high-frequency rfid tokens. In: *4th Workshop on RFID Security (RFIDSec)*. [S.l.: s.n.], 2008. p. 100–113.

7 HASELSTEINER, E.; BREITFUSS, K. Security in near field communication (nfc). In: *Workshop on RFID security*. [S.l.: s.n.], 2006. p. 12–14.

8 MADHAVAN, S. et al. *Wireless communication using compact certificates*. Google Patents, 2012. US Patent 8,327,146. Disponível em: <http://www.google.com.ar/patents/US8327146>.

9 PN532 User Manual at http://www.nxp.com/documents/user_manual/141520.pdf. In: . [s.n.]. Disponível em: <http://www.nxp.com/documents/user_manual/141520.pdf>.

10 TIMALSINA, S. K.; BHUSAL, R.; MOH, S. Nfc and its application to mobile payment: Overview and comparison. In: IEEE. *Information Science and Digital Content Technology (ICIDT), 2012 8th International Conference on*. [S.l.], 2012. v. 1, p. 203–206.