Kismet como Ferramenta de Análise de Rede

Flávio Gomes Figueira Camacho
Diretoria de T.I.
Vipnet Baixada Telecomunicações e Informática LTDA
Rua José de Alvarenga, 265
25.020-140 - Duque de Caxias - RJ - Brasil
+55 21 3799-8299
flavio@vipnettelecom.com.br

Luiz Claudio Schara Magalhães
Depto de Eng. de Telecomunicações
Universidade Federal Fluminense
Rua Passo da Pátria 156
24.220-121 - Niterói – RJ – Brasil
+55 21 2629-5696
schara@midiacom.uff.br

Resumo

Abordamos a utilização do Kismet como ferramenta de analise de rede sem fio, e como o mesmo pode ser utilizado para diagnosticar a causa de comportamentos inesperados em redes wireless, como o caso da Faculdade de Engenharia da Uff, com muitos clientes e pouca banda consumida.

***Palavras-chave* — Wi-Fi, Kismet, Redes, 802.11, Segurança, OpenWRT.**

I - INTRODUÇÃO

Foi observado através de gráficos gerados pelo MRTG que no hall de entrada da Faculdade de Engenharia da UFF, eram observados muitos clientes conectados a internet por um ponto de acesso que se encontra no local, mas o consumo de banda era muito pequeno, este comportamento não era esperado e este artigo busca responder o que estaria acontecendo, para isso decidiu-se colocar um segundo radio ao lado do que apresentava este comportamento, e substituir o seu *firmware* pelo OpenWRT que é uma versão de Linux, nele compilou-se o Kismet, que é um software de monitoramento passivo, ou seja, que não interfere em nada no ambiente sobre estudo, para tentar descobrir o motivo de tal comportamento. O Kismet ficou no mesmo canal do AP capturando todos os pacotes que trafegavam naquele ambiente, para que nós analisássemos o trafego e pudéssemos determinar a causa de tal comportamento.

II – KISMET

Kismet é um detector de redes, farejador (sniffer) de pacotes, e sistema de detecção de intrusão para redes sem fio 802.11, ele funciona com qualquer placa wireless que suporta o modo de monitoramento, e pode farejar tráfego 802.11a, 802.11b , 802.11g e 802.11n . O programa é executado em Linux, OpenWRT, FreeBSD, NetBSD, OpenBSD e Mac OS X. O cliente também pode ser executado no Microsoft Windows. É um software livre sendo distribuído sob a GNU (General Public License).

Ele é diferente da maioria dos outros detectores de rede sem fio pois ele trabalha de forma passiva. Isto significa que, sem enviar pacotes, ele é capaz de detectar a presença tanto dos pontos de acesso sem fios quanto dos clientes sem fio, e associá-los um com o outro. Também inclui as funções básicas da IDS sem fio, como a detecção de programas farejadores sem fio ativos como o NetStumbler, bem como uma série de ataques de rede sem fio.

O Kismet tem a capacidade de registrar todos os pacotes capturados e salvá-los em um formato de arquivo compatível com Airsnort tcpdump e Wireshark.

Outra caracteristica interessante é a sua capacidade de detectar solicitações de sondagem padrão e determinar o nível de criptografia sem fio usado em um determinado ponto de acesso.

Para encontrar o maior número de redes possível, Kismet suporta salto de canal. Isso significa que ele pode mudar constantemente de canal para canal de forma não sequencial, em uma seqüência definida pelo usuário com um valor padrão que deixa grandes buracos entre os canais (por exemplo, 1-6-11-2-7-12-3-8-13-4-9-14-5-10). A vantagem deste método é que ele irá capturar mais pacotes porque os canais adjacentes se sobrepoem .

Kismet também suporta o registro de coordenadas geográficas da rede se a entrada de um receptor de GPS estiver disponível .

Kismet tem três partes separadas.

- Um robô (drone) pode ser usado para coletar os pacotes e, em seguida, passá-los para um servidor para interpretação.
- Um servidor que pode ser usado sozinho ou em conjunto com um robô, atua na interpretação, organização e extrapolação dos pacotes capturados.
- O cliente que se comunica com o servidor e exibe as informações do servidor de coleta .

III - OPENWRT

O OpenWRT é uma distribuição Linux para Sistemas Embarcados, em que seu desenvolvimento dispõe de um sistema de arquivos totalmente escrito com gerenciamento de pacotes, e não apenas um *firmware* simples com um único processo, isso propicia uma grande facilidade na seleção e configuração das aplicações fornecidas e permite que o colaborador/usuário customize o dispositivo usando apenas pacotes. Para o colaborador o OpenWRT é a estrutura ideal para construir uma aplicação sem ter a preocupação de

construir um *firmware* completo em torno dela. Para o usuário isto significa a habilidade de total customização, para utilizar o dispositivo de maneira nunca antes vista.

Com a liberação dos códigos fontes Linux da série de roteadores *Linksys* WRT54G/GS surgiram um grande número de *firmwares* modificados para estender as funcionalidades em várias maneiras. Esses *firmwares* modificados continham 99% de códigos conservados e 1% de funcionalidades adicionais, e cada projeto tentou seguir um determinado segmento de mercado com as funcionalidades que forneceu.

Abaixo duas modificações para inicialização do projeto OpenWRT:

- Era freqüentemente difícil encontrar um *firmware* com a combinação da funcionalidade desejada;
- Todos os *firmwares* foram baseados nas fontes originais da *Linksys* que seguiram o desenvolvimento GNU/Linux.

O desenvolvimento do OpenWRT se deu de forma diferente, ao invés de começar a partir dos fontes do *Linksys*, o desenvolvimento começou em um estado limpo dos códigos fontes, com apenas códigos fontes base. Peça por peça o *software* foi montado trazendo as funcionalidades de volta, como no *firmware* padrão da *Linksys*, usando versões recentemente disponíveis.

Com o OpenWRT os pequenos dispositivos tornam-se mini PC Linux, possuindo praticamente todos os comandos Linux tradicionais e com um sistema de gerenciamento de pacotes para facilmente carregar um *software* ou características extra.

Porque utilizar o OpenWRT?

Porque o GNU/Linux nos dá o poder de fazer o que nós necessitamos com ferramentas baratas e evitar *software* proprietário, de código fechado. O OpenWRT é o *firmware* baseado no Linux mais rápido disponível para vários roteadores sem fio. Alem disso, a comunidade de OpenWRT fornece os mais variados pacotes *add-on*, como o Kismet que citamos anteriormente.[1]

IV – CENÁRIO

O estudo foi feito no Hall de entrada da Faculdade de Engenharia da Universidade Federal Fluminense, onde está disponível um Ponto de Acesso sem fio a Internet, que registrava através do MRTG muitos usuários mas pouca banda consumida, o que causava uma duvida do porque tal comportamento, nosso objetivo era tentar determinar o motivo deste comportamento.

V – METODOLOGIA

Para efetuar nosso estudo foi utilizado um rádio TP-Link WR740N escolhido por seu baixo custo e facilidade de ser encontrado no mercado. Substituímos o firmware padrão de fábrica pelo OpenWRT, que é um sistema operacional baseado no kernel do Linux, e dentro deste foi instalado o pacote Kismet-drone versão 2010-07-R1-2 com suas dependências uclibcxx, libni-tiny, libcap e libpcre, que foi configurado para capturar os pacotes pela interface sem fio e enviá-los através da interface Ethernet para um PC com CentOS 6.3 e o pacote do Kismet-server **(**kismet-3.0.1-201007r1.1.el6.rf.i686.rpm) instalado.

Instalamos este rádio ao lado do AP que fornece internet na entrada do prédio da faculdade de engenharia, e deixamos o mesmo capturando pacotes durante um dia inteiro e enviando os mesmos para o servidor que ficou armazenando os pacotes capturados.

O Kismet é um analisador de rede (Sniffer) funcionando como um Scanner. Ele consegue detectar pontos de acesso, e capturar seus pacotes. É utilizado como verificador de segurança ou localizador de redes próximas, assim sendo, poder ser usado para o bem ou para o mal. Ele é um recurso passivo, que uma vez ativado coloca a placa de rede sem fio no modo rfmon, capturando os sinais que chegam a antena. Assim, mesmo que os pontos de acesso tenham sido configurados para não divulgar o ESSID ou com criptografia eles vão ser detectados. Algo que não é possível com outras ferramentas como o Netstumbler por exemplo, já que estes pontos de acesso não respondem a pacotes de broadcast, já o Kismet consegue detectá-los visto que na conexão do cliente o ESSID é transmitido de forma não criptografada no processo de associação do cliente ao ponto de acesso. Primeiramente, essa rede será detectada como "no ssid", já que o broadcast do SSID foi desativado no ponto de acesso. Mas, assim que qualquer computador se conecta ao ponto de acesso, o Kismet descobre o SSID correto. Como ele não transmite pacotes, apenas escuta as transmissões, todo o processo é feito sem prejudicar as redes vizinhas, de forma praticamente indetectável.

O kismet gera os seguintes arquivos de log.

- .alert – Alertas gerados pelo IDS
- .gpsxml – Posicionamento de GPS
- .nettxt – Redes em TXT
- .netxml – Redes em XML
- .pcapdump – Todos os pacotes capturados

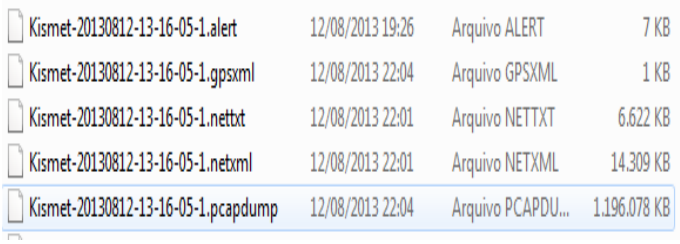

Fig. 1 - Arquivos de log do Kismet que foram estudados.

Estes arquivos foram exportados para uma maquina Windows com Wireshark e Excel onde fizemos a análise dos dados coletados.

VI – RESULTADOS

A. arquivo .alert

No período de análise foram observados 34 alertas sendo:

ADHOCCONFLICT – 10 ocorrências

Redes Ad hoc com o mesmo BSSID 00:00:00:00:00:00

BCASTDISCON – 2 ocorrências

Atacante desconectando clientes de uma rede, provocando uma denial-of-service, que dura apenas enquanto o atacante é capaz de enviar os pacotes.

CHANCHANGE – 3 ocorrências

Um ponto de acesso previamente detectado mudar de canal pode indicar um ataque de spoofing. Por spoofing um AP legítimo em um canal diferente, um atacante pode atrair clientes para o falsificado ponto de acesso. Uma mudança de canal AP durante a operação normal pode indicar que um ataque está em andamento, no entanto centrais de redes gerenciadas podem mudar automaticamente os canais AP para áreas menos utilizadas do espectro.

CRYPTODROP – 1 ocorrência

Falsificar um AP com opções de criptografia menos seguras pode enganar clientes em conexão com credenciais comprometidas. A única situação em que um ponto de acesso deve reduzir criptografia é quando a segurança do AP é reconfigurada.

DEAUTHCODEINVALID – 14 ocorrências

A especificação 802.11 define os códigos de razão válidos para eventos de desconecção e deautenticação.

Todos os MACs eram inválidos tanto do AP, quanto do cliente.

DISCONCODEINVALID - 4 ocorrências

A especificação 802.11 define os códigos de razão válidos para eventos de desconecção e deautenticação.

Todos os MACs eram inválidos tanto do AP, quanto do cliente.

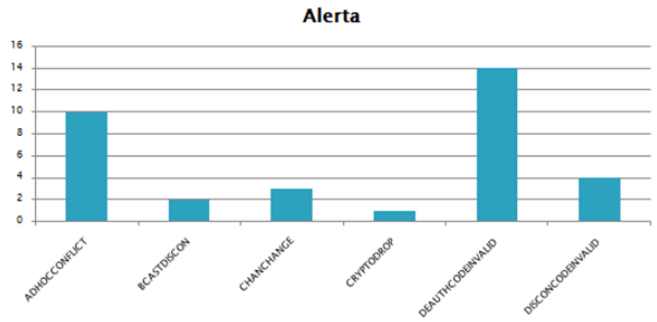

Fig. 2 - Quantidades de alertas IDS do Kismet por tipo.

B. arquivo .gpsxml

Não havia informações neste arquivo pois o dispositivo não tinha GPS.

C. arquivo .netxt

Arquivo que contém os logs de rede no formado txt, utilizamos na análise o arquivo *.netxml.*

D. arquivo .netxml

Arquivo que contem os logs de rede no formado xml. Este arquivo foi exportado para o Excel para analisar seus dados e gerar os gráficos abaixo.

Quantidade de pacotes capturados = 117.695

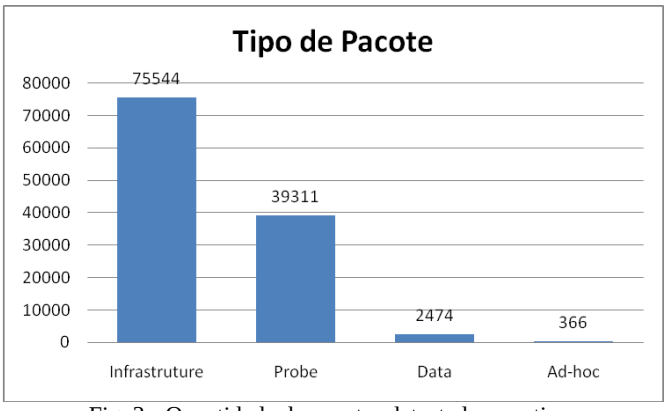

Fig. 3 - Quantidade de pacotes detectados por tipo.

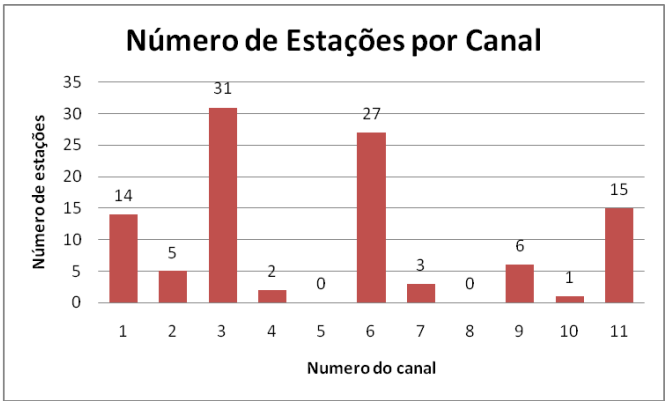

Fig. 4 - Número de estações por canal

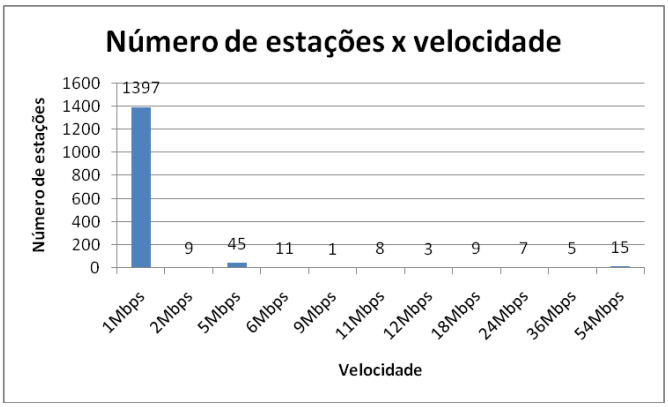

Fig. 5 - Número de estações por velocidade

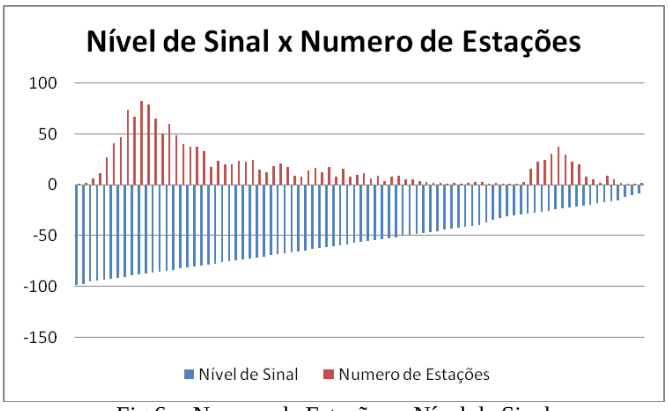

Fig 6. - Numero de Estações x Nível de Sinal

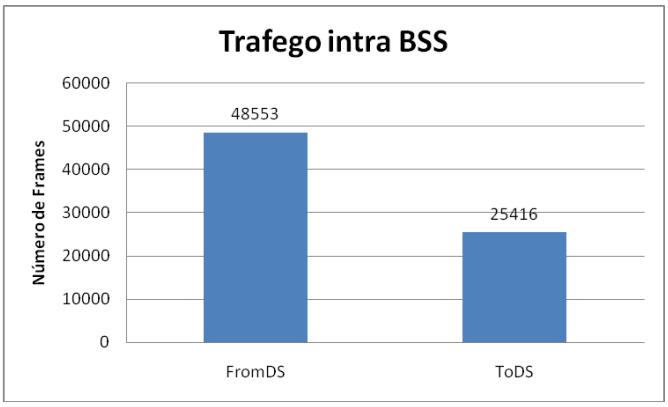

Fig. 7 - Trafego intra BSS

- Número de SSID observados = 880

E. arquivo .pcapdump

Arquivo que contem os cabeçalhos dos pacotes capturados. Este arquivo foi exportado para um PC Windows

e aberto com o Wireshark, onde foi feita análise dos dados apresentados abaixo.

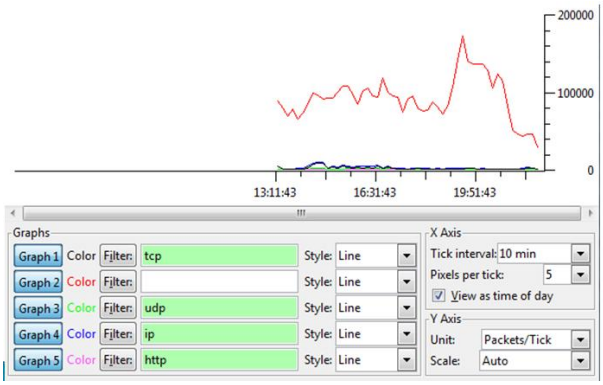

Fig. 8 - Gráfico com a quantidade de pacotes capturados dos protocolos TCP(preto), UDP(verde), IP(azul), HTTP(rosa) e outros(vermelho), em função do tempo.

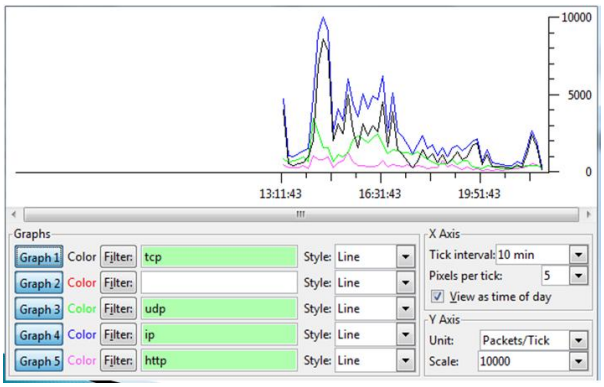

Fig. 9 - Gráfico com a quantidade de pacotes capturados dos protocolos TCP(preto), UDP(verde), IP(azul) e HTTP(rosa), em função do tempo.

Protocol	% Packets	Packets	% Bytes	Bytes
Frame	100,00 %	4905166	100,00 %	1146300182
PPI Packet Header	100,00 %	4905166	100,00 %	1146300182
IEEE 802.11 wireless LAN	100,00 %	4905166	100,00 %	1146300182
IEEE 802.11 wireless LAN management frame	27,63 %	1355274	25,46 %	291850048
Malformed Packet	0,02 %	1210	0,02 %	271850
Data	23,98 %	1176446	61,24 %	701490108
Logical-Link Control	3,61 %	177076	4,50 %	51546946
Internet Protocol Version 4	2,83 %	138785	3,93 %	45058500

Fig. 10 - Porcentagem de pacotes por tipo.

VII – ANÁLISE DOS DADOS CAPTURADOS

O arquivo .alert apresentou 34 alertas, em seis diferentes categorias, sendo que todas as falhas foram analisadas e mostravam situações suspeitas mas que não eram reais problemas de segurança.

O arquivo .gpsxml não apresentou dados pois não utilizamos gps acoplado ao AP.

Os arquivos .netxt e .netxml apresentam a mesma informação das redes encontradas, sendo que um em formato txt e outro em xml, exportamos o arquivo xml para o Excel onde analisamos os dados com a confecção dos gráficos apresentados, que nos trouxeram informações muito valiosas a respeito do que estava ocorrendo na rede Wireless do hall de entrada da Faculdade de Engenharia.

Este arquivo tinha o tamanho de aproximadamente 14Mb, e continha 117.695 pacotes capturados, sendo 75544 de infraestrutura (64%), a surpresa ficou por conta do número muito alto de pacotes de probe (33%), e pela presença de uma pequena quantidade de pacotes de ad-hoc (0,3%), indicando um comportamento inesperado, onde os alunos não se conectam apenas com o AP, mas estabelecem conexões diretas entre seus dispositivos mesmo que seja uma atividade muito pequena.

A análise do número de estações por canal também trouxe uma informação interessante, ha uma grande aglomeração de APs nas mesmas freqüências enquanto outras ficam sem nenhum uso, os canais 5 e 8 não apresentavam qualquer estação enquanto o canal 3 possuía 31 estações diferentes, ou seja, o espectro eletromagnético não estava sendo utilizado de forma eficiente, o que contribui para a baixa eficiência do canal, e para a baixa taxa da dados agregada, pois o piso de ruido ficando muito alto obriga o AP a utilizar modulações mais robustas, que sacrificam a taxa de dados em função da estabilidade.

O estudo do numero de estações em função da velocidade mostra que tínhamos 1397 estações trabalhando na velocidade de 1Mbps enquanto apenas 15 conseguiram chegar a velocidade de 54Mbps, logo possuímos muitos clientes, só que a velocidade agregada de cada um era muito baixa, o que ajuda a justificar o causa de tantos usuários e um consumo muito baixo de banda, na verdade a velocidade disponível era muito baixa.

A figura 6 relaciona o numero de clientes e o sinal dos mesmos, observa-se claramente, que a grande maioria dos clientes está conectado com um sinal muito fraco menor do que -70dBm, o que justifica o AP modular a velocidades muito baixas. Temos um outro grupo de clientes com sinal muito forte, o que se deve ao usuário se aproximar ao máximo do AP na tentativa de melhorar a sua conexão. Muitos usuários são clientes que se conectam via celular, que mesmo não estando trafegando dados permanecem conectados ao AP, aumentando o número de usuários conectados mas com baixo tráfego, alem disso muitos celulares estão dentro de bolsas de mulher, mochilas... o que faz com que o seu sinal seja baixo e reduza a modulação dos clientes como um todo.

O trafego intraBSS também é um fato inesperado, imaginávamos que os alunos se conectavam ao AP para acessa a Internet e não pra trocar arquivos entre si, mas o gráfico da figura 7 demonstra que é muito freqüente a comunicação intraBSS.

Outra observação interessante é que ouvimos tentativas de conexão para 880 SSID diferentes, o que demonstra que os dispositivos tentam se conectar ativamente as redes cadastradas nas interfaces de rede, e um observador passivo como o Kismet é capaz de capturar isso, podendo determinar as localidades por onde o cliente costuma se conectar.

O arquivo .pcapdump tem os cabeçalhos de todos os pacotes capturados, o que gerou um arquivo de 1.2Gb em apenas 8hs de monitoramento, e que foi exportado para uma maquina windows e aberto com o wireshark, nos fornecendo os dados observados, como o da figura 8 onde demonstra que de todos os pacotes capturados o trafego IP,TCP,UDP e HTTP representa um volume muitíssimo pequeno do total.

Já na figura 9 restringimos o gráfico apenas a estes quatro protocolos e observamos o trafego HTTP representa uma parcela pequena da comunicação total.

Na figura 10 observamos que 27% dos pacotes, ou seja, mais de 1 em cada 4, são frames de gerência de rede sem fio, ocupando uma banda de 25%.

VIII – CONCLUSÕES

Conseguiu-se demonstrar a utilidade e flexibilidade da ferramenta Kismet na analise de redes wireless através da captura passiva dos pacotes que trafegam entre o AP e a estação, alem de apresentar uma hipótese do motivo do comportamento inesperado onde tínhamos muitos clientes e pouca banda consumida, na verdade tínhamos muitos clientes, só que conectados a uma velocidade máxima de 1Mbps, provavelmente por estarem com um sinal muito baixo.

REFERÊNCIAS

[1] F. L. Deboni, and R. F. Borba, "Sistemas Embarcados em Segurança de Redes - OpenWRT", Faculdade Salesiana de Vitória, 2007.

[2] K. Chintalapudi, A. Iyer, and V. Padmanabhan, "Indoor localization without the pain", In Proc. of ACM MobiCom, 2010

[3] E. Aryafar, N. Anand, T. Salonidis, and E. Knightly, "Design and experimental evaluation of multi-user beamforming in wireless LANs", In Proc. of ACM MobiCom, 2010.

[4] L. Hyuk, L. Kung, J. C. Hou, and H. Luo, "Zero-configuration indoor localization over IEEE 802.11 wireless infrastruture", Journal Wireless Networks, vol. 16, pp. 405-420, Fevereiro 2010.

[5] L. Hyuk, L. Kung, J. C. Hou, and H. Luo, "Zero-configuration, Robust Indoor Localization: Theory and Experimentation", In Proc. of IEEE Infocom, 2006.

[6] C. Wu, Z. Yang, Y. Liu, and W. Xi, "Will: Wireless Indoor Localization Without Site Survey", Transaction On Parallel and Distributed System, vol X, No.X, Fevereiro 2012

[7] F. Fainelli, "The OpenWRT embedded development framewok", www.openwrt.org, 2008

[8] K. Sumit, "Online monitoring using kismet" Master Projects, pp 243, San Jose State University, 2012.

[9] R.C. Carrano, R. R. Martins, L. C. Schara Magalhães, "The RUCA project and Digital Inclusion", Network Operations and Management Symposium, 2007. LANOMS 2007.

[10] N. Patwari and S. Kasera. "Robust location distinction using temporal link signatures", In Proc. of the ACM ModiCom Conf., pp 111-122, Set. 2007.

[11] L.Chappell, "Wireshark Network Analysis", Protocol Analysis Institute, Jul 2012

[12] http://www.kismetwireless.net/documentation.shtml

[13] http://luci.subsignal.org/trac

[14] http://openwrt.org/

[15] http://support.linksys.com/pt-latam/gplcodecenter

[16] http://www.midiacom.uff.br/scifi

[17] http://www.wireshark.org/