Detecção de APs Falsos Utilizando TCP-RTT
Relatório de Acompanhamento - SAAP

Felipe J. Chaulet

[1]Centro de Informática – Universidade Federal de Pernambuco (CIn/UFPE)
Av. Jornalista Anibal Fernandes, s/n – Cidade Universitária – 50.740-560 – Recife – PE

fjc@cin.ufpe.br

1. Introdução

O crescimento do uso de redes sem fio móveis aconteceu exponencialmente nos últimos anos, em virtude da baixa nos preços dos equipamentos eletrônicos que possuem interface Wi-Fi, tendo como consequência a sua popularização e a disponibilização de comunicação sem fio em locais públicos como shopping centers, aeroportos, lojas de conveniência e hotéis. Em muitos casos, as redes disponíveis nestes locais não possuem segurança baseada em criptografia, pois isso demanda o trabalho de configurar todos os dispositivos da rede com uma chave pré-compartilhada ou a instalação de um servidor de autenticação, trazendo custos tanto financeiros, quanto de mão-de-obra especializada.

Essa atitude oferece várias oportunidades para invasores utilizarem os recursos da rede para obter informações importantes dos usuários, como informações de autenticações em serviços online e informações de bancárias. Isso acontece porque as redes sem fio não possuem um meio físico para a transmissão de dados e dessa forma, qualquer usuário com um dispositivo compatível com o equipamento instalado na rede, pode capturar informações da mesma. Para resolver esse problema, a segurança é aplicada em nível de aplicação, utilizando protocolos que aplicam criptografia na transação das informações trocadas na rede.

É comum em redes sem fio, a presença de pontos de acesso falsos, também conhecidos por Rogue APs ou Fake APs, que estão presentes em cerca de 20% das redes corporativas, de acordo com [1], tornando-se uma grande ameaça para os usuários, pois através deles e possível capturar informações enviadas pelos usuários sem que eles percebam.

2. Trabalhos Relacionados

Esta seção apresenta os trabalhos relacionados ao tema do trabalho, trazendo os métodos de detecção de pontos de acessos falsos que serviram como base para a elaboração da proposta apresentada na Seção 4. Os métodos de detecção de pontos de acessos falsos podem ser divididos em duas categorias, sendo os métodos que são do lado do administrador, ou seja, o administrador da rede é o responsável por administrar uma estrutura complementar à rede capaz de detectar os pontos de acesso falsos e assim, eliminá-los. Já na outra categoria, conhecida como lado do cliente, parte do pressuposto de que o cliente utiliza algum recurso para identificar se o ponto de acesso que ele está utilizando para acessar a rede é verdadeiro ou falso.

Dentre as inúmeras soluções pertencentes às categorias supracitadas, algumas estão explanadas nas sub-seções subseqüentes.

2.1. Iniciativa do Administrador da Rede (*adm-side*)

As abordagens que partem do administrador da rede ou *adm-side*, são abordagens que partem da equipe responsável pela administração da rede. Esse tipo de abordagem é interessante para redes corporativas, SOHO (Small Office, Home Office) e pessoais, pois exige uma participação mais intensa do administrador da rede.

Boa parte das abordagens *adm-side* realizam algumas mudanças na estrutura da rede, seja com a instalação de novos dispositivos ou na alteração dos existentes para fornecer as informações necessárias para a detecção de Fake APs e trazem como consequência dessas alteraçoes, em alguns casos, um alto custo.

A grande maioria das abordagens existentes é *adm-side* e envolvem o uso de recursos como *fingerprinting*, que consiste, basicamente, na identificação única de um dispositivo da rede. Os trabalhos [2], [3], [4], [5], [6] e [7] utilizam essa técnica para a detecção de pontos de acessos falsos.

Outro método utilizado é reconhecendo características de tráfego, para assim detectar alguma anormalidade na rede e identificar pontos de acessos falsos. Essa técnica é utilizada em [8], [9], [10], [11], [12], [13], [14] e [15].

A utilização de medições relacionadas com o tempo também é bastante utilizada, como por exemplo, o tempo de chegada entre pacotes, apresentada em [16] e que consiste em utilizar um servidor capaz de diferenciar os saltos usados pelos pacotes de um determinado cliente, calculando o tempo que cada pacote leva pra sair de sua origem, chegar ao destino e voltar a sua origem. O trabalho [17], utiliza uma métroca semelhante, porém tem como objetivo identificar um ap falso conectado à rede via cabo, utilizando uma medida de tempo semelhante, chamada RTT (Round-Time Trip). Neste segundo trabalho, o objetivo é identificar se existe comunicação sem fio em um ponto da rede onde deveria existir apenas comunicação via cabo.

Conforme mostrado nas abordagens anteriores, algumas delas consistem em capturar dados da rede para análises adicionais, consistidas por uma metodologia e oferecendo uma forma de diferenciar aps verdadeiros e aps falsos. Os trabalhos [15] e [18], realizam esse processo de coleta de informações implantando sensores para monitorar o tráfego da rede e fornecer dados para análise. Dentre as desvantagens desse método está o alto custo financeiro, caso a área na qual os sensores devem ser instalados for muito grande e a mesma pode gerar um grande tráfego de informações dos sensores para um servidor central, que torna-se detentor de todas essas informações e pode acabar sendo algo de ataques também. Outras soluções que utilizam sensores são apresentadas em [19], [20] e [21].

2.2. Iniciativa do Usuário da Rede (*user-side*)

Uma vantagem das abordagens *user-side* é que um usuário qualquer pode utilizar o método para identificar se o AP que está utilizando para acessar a rede é verdadeiro ou não, sem requisitar um grande conhecimento de conceitos técnicos sobre segurança e redes sem fio.

Outro ponto inportante de salientar é que as abordagens *adm-side* oferecem informações aos administradores da rede, permitindo que os mesmos detectem e eliminem os Fake APs da rede sem o conhecimento do usuário. Embora esse ponto possa ser

visto como uma solução que é completamente transparente ao usuário, ele não tem uma confirmação da segurança desse método, em outras palavras, o usuário não possui uma informação visual que indique a segurança da conexão. Já a abordagem *user-side*, executada pelo próprio usuário, oferece uma informação visual, deixando claro para o cliente a situação do meio de acesso que ele está utilizando.

Abordagens *user-side*, por não exigir nenhuma modificação na rede, utilizam informações que podem ser acessadas diretamente pelo usuário, como por exemplo, o estudo [22] que utiliza características de tráfego, através do RTT. O uso do RTT também está presente em abordagens *adm-side*, conforme apresentado em [23]. A desvantagem da abordagem de [22] é o uso de ferramentas que tem difícil instalação e gerenciamento e que necessitam de uma versão específica do kernel do sistema. Assim, apenas um usuário que tenha conhecimentos profundos sobre compilação de kernel seria capaz de utilizar a ferramenta.

3. Problema

O problema a ser tratado neste trabalho, envolve um cenário onde um usuário malicioso cria um ponto de acesso falso para fazer com que outros usuários da rede conectem-se a ele ao acessar algum recurso da rede. Assim o usuário malicioso pode ter acesso às informações enviadas pelos usuários da rede. O ponto de acesso falso, neste cenário, consiste de um computador com duas interfaces de rede sem fio, sendo que através de uma delas, o usuário maliciosos irá conectar-se ao ponto de acesso legítimo da rede e a outra interface de rede irá operar como se fosse um ponto de acesso e será através dela que os usuários irão conectar-se ao ponto de acesso falso.

Com o objetivo de tornar o ponto de acesso falso mais difícil de ser identificado, o usuário malicioso poderá configurá-lo de forma que ele terá as mesmas características do ponto de acesso verdadeiro. Essas características são informações que o ponto de acesso possui que permitem sua identificação, como por exemplo o seu SSID (Service Set Identification, o BSSID (Basic Service Set Identification), o canal, o tipo de criptografia que o mesmo utiliza, dentre outras. Essas informações ficam acessíveis a qualquer dispositivo que estiver dentro da área de cobertura do ponto de acesso e ele as envia periodicamente através de pacotes *beacom* enviados em *broadcast*. A Figura 1 mostra o cenário supracitado.

O motivo de configurar o ponto de acesso falso com as mesmas características do ponto de acesso verdadeiro é que atualmente os algoritmos de seleção de AP utilizam a força de sinal (RSS - Received Signal Strenght) para definir em qual ponto de acesso conectar. Isso implica que, caso no raio de alcance de um dispositivo estiverem dois pontos de acesso com as mesmas credenciais (SSID, BSSID, canal e criptografia), o AP selecionado será o que possuir o maior RSS. Assim, configurando o ponto de acesso falso com as mesmas credenciais que o ponto de acesso verdadeiro, o atacante apenas precisa mover o ponto de acesso falso para um local próximo ao usuário que ele quer atacar.

É importante salientar que no cenário adotado, não haverá intervenção da administração da rede e por esse motivo a solução adotada deverá possuir uma iniciativa do cliente. Portanto, o problema a ser resolvido é encontrar uma maneira de fazer cliente identificar o ponto de acesso falso.

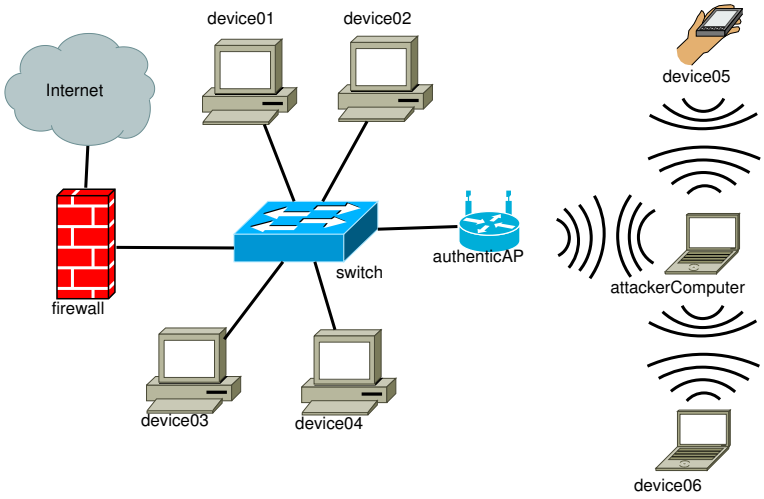

Figura 1. *Exemplo de um Fake AP implantado por um usuário malicioso*

4. Proposta

Para solucionar o problema apresentado na Seção 3, será utilizado o TCP-RTT, que é uma medida de RTT realizada sobre o protocolo TCP. Nessa medida, é calculado o tempo que um pacote leva para sair de seu emissor, chegar a seu destinatário e a resposta a esse pacote voltar ao seu emissor. O atacante, entretanto, pode prever o valor do RTT, caso o mesmo seja calculado entre dois hosts da rede, visto que o resultado do RTT vai variar apenas de acordo com o congestionamento da rede. Para evitar esse cenário, será realizada uma requisição para um host na Internet. Assim, visto que o atacante não saberá qual o host será consultado, ele não pode prever o valor do RTT. Porém, a Internet é instável e o host consultado pode possuir uma conexão baixa e afetar os valores do RTT. Para resolver este problema, serão utilizados hosts confiáveis, ou seja, websites de grandes corporações que possuem grande disponibilidade.

O RTT é uma medida válida, visto que quando o usuário está conectado diretamente ao ponto de acesso verdadeiro, um pacote enviado pelo mesmo sairá de sua origem, passando pelo ponto de acesso verdadeiro, sendo encaminhado para a Internet e chegando em seu destino. Já quando o usuário está conectado na rede através de um ponto de acesso falso, o trajeto do pacote envolve a saída de sua origem, sendo receptado pelo ponto de acesso falso, que por sua vez reencaminha o pacote para o ponto de acesso verdadeiro que envia o pacote para seu destino, através da Internet. Portanto, o objetivo da solução é utilizar medidas de TCP-RTT para diferenciar o tempo consumido pelo pacote ao passar pelo ponto de acesso falso.

5. Metodologia

Para alcançar a proposta presente na Seção 4, um cenário de testes foi construído, permitindo assim, o uso do mesmo para a realização de testes e geração de resultados.

A solução foi desenvolvida em ambiente GNU/Linux x86, utilizando-se de um computador com duas interfaces de rede, fazendo o papel de ponto de acesso falso, através da suíte de aplicativos aircrack-ng, o qual possui a ferramenta airbase-ng, capaz de fazer com que uma placa de rede sem fio compatível opere em modo de ponto de acesso. Já

outro computador, também em ambiente GNU/Linux x86 fará o papel de cliente, sendo que o mesmo executará um código capaz de enviar requisições para hosts de alta disponibilidade na Internet e armazenar os valores do TCP-RTT dessas requisições.

Os requisitos necessários para a execução da solução são:

- Computador com sistema operacional GNU/Linux;
- Acesso de super usuário;
- Tshark, versão em modo texto do aplicativo Wireshark;
- Curl, biblioteca que permite a realização de requisições HTTP;
- Estar conectado ao ponto de acesso;
- Aproximadamente 20 megabytes de espaço em disco, para armazenamento de informações.

6. Implementação

A captura de informações da rede foi implementada através da linguagem Shell Script, já presente nas distribuições GNU/Linux, e está representada no Algoritmo 6.1. O envio das requisições HTTP foi escrito na liguagem C utilizando a biblioteca Curl. A liguagem C também foi utilizada para realizar outros cálculos.

Algoritmo 6.1 Pseudo Código do Algoritmo de Captura de Informações da Rede

```
1:  urls ← urls.txt;
2:  ips ← urls.convert();
3:  filter ← ips.concatenate();
4:  network.SSID ← info.SSID();
5:  network.BSSID ← info.BSSID();
6:  network.CHANNEL ← info.channel();
7:  while network ← data.capture() do
8:      HTTPRequests.send;
9:  data.filter ← filter;
10: storage ← filteredData;
```

A solução conta com um arquivo chamado *"urls.txt"* que contém os endereços dos websites para os quais as requisições HTTP serão enviadas (linha 1). Em virtude do fato de alguns websites possuirem mais de um servidor web em endereços de IPs diferentes e que por ventura alguma requisição possa ser redirecionada, todos os endereços de IP dos websites são armazenados e concatenados para compor um filtro, com o objetivo de manipular apenas os pacotes destinados e oriundos destes endereços (linhas 2 e 3). Algumas informações da rede precisam ser capturadas, para a identificação da mesma (linhas 4, 5 e 6). Durante o tempo no qual as requisições HTTP estiverem sendo executadas, é necessário que os dados da rede estejam sendo capturados, para que as requisições e respostas destas requisições sejam armazenadas para análise (linhas 7 e 8). Por fim, o filtro construído na linha 3 é utilizado para separar as informações importantes (linha 9) e assim, os resultados são salvos em arquivo (linha 10).

7. Cronograma

Esta seção apresenta o cronograma para as tarefas ainda faltantes para o término do desenvolvimento da pesquisa aqui apresentada. Para melhor visualizar o agendamento de tarefas, a Tabela 1 apresenta as atividades para os próximos 6 meses, considerando as atividades dispostas a seguir.

Tabela 1. Cronograma de Atividades

Atividade	Março	Abril	Maio	Junho	Julho	Agosto
1	x	x				
2	x	x				
3	x	x	x			
4		x	x	x	x	
5						x

- Atividade 1: Realização de testes;
- Atividade 2: Correções de possíveis erros e realização de possíveis otimizações;
- Atividade 3: Coleta de resultados;
- Atividade 4: Elaboração dissertação;
- Atividade 5: Defesa.

Referências

[1] L. Ma, A. Teymorian, X. Cheng, and M. Song, "RAP: Protecting Commodity Wi-Fi Networks from Rogue Access Points," in *Proceedings of the Fourth International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness & Workshops*. ACM, 2007, pp. 1–7.

[2] S. Jana and S. Kasera, "On Fast and Accurate Detection of Unauthorized Wireless Access Point Using Clock Skews," *Mobile Computing, IEEE Transactions on*, vol. 9, no. 3, pp. 449–462, 2010.

[3] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless Device Identification with Radiometric Signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking (MobiCom 2008)*, 2008, pp. 116–127.

[4] C. Corbett, R. Beyah, and J. Copeland, "A Passive Approach to Wireless NIC Identification," in *Proceedings of the IEEE International Conference on Communications (IEEE/ICC)*, vol. 5. IEEE, 2006, pp. 2329–2334.

[5] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. Randwyk, and D. Sicker, "Passive Data Link Layer 802.11 Wireless Device Driver Fingerprinting," in *Proceedings of the 15th USENIX Security Symposium*, 2006, pp. 167–178.

[6] K. Gao, C. Corbett, and R. Beyah, "A Passive Approach to Wireless Device Fingerprinting," in *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2010, pp. 383–392.

[7] D. Takahashi, Y. Xiao, Y. Zhang, P. Chatzimisios, and H.-H. Chen, "IEEE 802.11 User Fingerprinting and Its Applications for Intrusion Detection," *Computers and Mathematics with Applications*, vol. 60, pp. 307–318, 2010.

[8] L. Watkins, R. Beyah, and C. Corbett, "A Passive Approach to Rogue Access Point Detection," in *Proceeding of Global Telecommunications Conference (IEEE/GLOBECOM)*, 2007, pp. 355–360.

[9] S. Shetty, M. Song, and L. Ma, "Rogue Access Point Detection by Analyzing Network Traffic Characteristics," in *Proceedings of the IEEE Military Communications Conference (IEEE/MILCOM)*, 2007, pp. 1–7.

[10] H. Yin, G. Chen, and J. Wang, "Detecting Protected Layer-3 Rogue APs," in *Proceeding of Fourth International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, 2007, pp. 449–458.

[11] W. Wei, S. Jaiswal, J. Kurose, and D. Towsley, "Identifying 802.11 Traffic from Passive Measurements using Iterative Bayesian Inference," in *Proceedings of the 25th Conference on Computer Communications (IEEE/INFOCOM)*, vol. 6. Citeseer, 2006.

[12] X. qiang Peng, C. Zhang, and D. gang Wang, "The Intrusion Detection System Design in WLAN Based on Rogue AP," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET)*, vol. 3. IEEE, 2010, p. V3.

[13] R. Beyah, S. Kangude, G. Yu, B. Strickland, and J. Copeland, "Rogue Access Point Detection Using Temporal Traffic Characteristics," in *Proceedings of the IEEE Global Telecommunications Conference (IEEE/GLOBECOM)*, vol. 4. IEEE, 2005, pp. 2271–2275.

[14] V. Baiamonte, K. Papagiannaki, and G. Iannaccone, "Detecting 802.11 Wireless Hosts from Remote Passive Observations," *NETWORKING 2007. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, pp. 356–367, 2010.

[15] P. Bahl, R. Chandra, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill, "Enhancing the Security of Corporate Wi-Fi Networks Using DAIR," in *Proceedings of the 4th international conference on Mobile systems, applications and services*, ser. MobiSys. ACM, 2006, pp. 1–14.

[16] Y. Song, C. Yang, and G. Gu, "Who is Peeping at Your Passwords at Starbucks??To Catch an Evil Twin Access Point," in *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2010, pp. 323–332.

[17] C. Mano, A. Blaich, Q. Liao, Y. Jiang, D. Cieslak, D. Salyers, and A. Striegel, "RIPPS: Rogue Identifying Packet Payload Slicer Detecting Unauthorized Wireless Hosts through Network Traffic Conditioning," *ACM Transactions on Information and System Security (TISSEC)*, vol. 11, no. 2, pp. 1–23, 2008.

[18] A. Adya, P. Bahl, R. Chandra, and L. Qiu, "Architecture and Techniques for Diagnosing Faults in IEEE 802.11 Infrastructure Networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking (MobiCom)*. ACM, 2004, pp. 30–44.

[19] U. Deshpande, T. Henderson, and D. Kotz, "Channel Sampling Strategies for Monitoring Wireless Networks," in *Proceedings of the 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*. IEEE, 2006, pp. 1–7.

[20] P. Bahl, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill, "DAIR: A Framework for Managing Enterprise Wireless Networks using Desktop Infrastructure," in *Proceedings of the Fourth Workshop on Hot Topics in Networking (HotNets)*. Citeseer, 2005.

[21] R. Chandra, A. Padhye, A. Wolman, and B. Zill, "A Location-based Management System for Enterprise Wireless Lans," in *Proceeding of USENIX Networked Systems Design and Implementation (NSDI)*, 2007, pp. 1–16.

[22] H. Han, B. Sheng, C. Tan, Q. Li, and S. Lu, "A Measurement Based Rogue AP Detection Scheme," in *Proceedings of the 28th Annual Joint Conference on Computer Communications (IEEE/INFOCOM)*, 2009, pp. 1593–1601.

[23] W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, and D. Towsley, "Passive Online Rogue Access Point Detection Using Sequential Hypothesis Testing with TCP ACK-pairs," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, 2007, pp. 365–378.

[24] C. Liu and J. Yu, "Rogue Access Point Based DoS Attacks Against 802.11 WLANs," in *Proceedings of the Fourth Advanced International Conference on Telecommunications (AICT)*. IEEE, 2008, pp. 271–276.

[25] L. Ma, A. Teymorian, and X. Cheng, "A Hybrid Rogue Access Point Protection Framework for Commodity Wi-Fi Networks," in *Proceedings of the 27th Conference on Computer Communications (IEEE/INFOCOM)*, 2008, pp. 1220–1228.

[26] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell, "Detecting 802.11 MAC Layer Spoofing Using Received Signal Strength," in *Proceedings of the 27th Conference on Computer Communications (IEEE/INFOCOM)*, 2008, pp. 1768–1776.

[27] V. Sriram, G. Sahoo, and K. Agrawal, "Detecting and Eliminating Rogue Access Points in IEEE-802.11 WLAN - a Multi-agent Sourcing Methodology," in *Proceedings of the IEEE 2nd International Advance Computing Conference (IEEE/IACC)*, 2010, pp. 256–260.

[28] A. Günther and C. Hoene, "Measuring round trip times to determine the distance between wlan nodes," *NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, pp. 303–319, 2005.

[29] H. Jiang and C. Dovrolis, "Passive estimation of tcp round-trip times," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 3, pp. 75–88, 2002.

[30] G. Li, N. Zhao, and C. Liu, "Round trip time estimation based on adaptive filtering," in *Information Science and Engineering (ICISE), 2009 1st International Conference on*. IEEE, 2009, pp. 1842–1846.

[31] G. Lusilao-Zodi, M. Dlodlo, G. De Jager, and K. Ferguson, "Round-trip time estimation in telecommunication networks using composite expanding and fading memory polynomials," in *MELECON 2010-2010 15th IEEE Mediterranean Electrotechnical Conference*. IEEE, 2010, pp. 1581–1585.

[32] Y. Pei, H. Wang, and S. Cheng, "A passive method to estimate tcp round trip time from nonsender-side," in *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*. IEEE, 2009, pp. 43–47.

[33] A. Saeed, S. Naseem, and Z. Zaidi, "Mobility estimation for wireless networks using round trip time (rtt)," in *Information, Communications & Signal Processing, 2007 6th International Conference on*. IEEE, 2007, pp. 1–5.

[34] B. Veal, K. Li, and D. Lowenthal, "New methods for passive estimation of tcp round-trip times," *Passive and Active Network Measurement*, pp. 121–134, 2005.

[35] H. Yan, K. Li, S. Watterson, and D. Lowenthal, "Improving passive estimation of tcp round-trip times using tcp timestamps," in *IP Operations and Management, 2004. Proceedings IEEE Workshop on*. IEEE, 2004, pp. 181–185.