Mobile IP

Cristina Morais da Silva
Departamento de Ciência dos Computadores,
Faculdade de Ciências da Universidade do Porto
Porto, Portugal
up201505454@edu.fc.up.pt

Rui Manuel Rodrigues dos Santos
Departamento de Ciência dos Computadores
Faculdade de Ciências da Universidade do Porto
Porto, Portugal
up201805317@edu.fc.up.pt

Sara Daniela Ferreira de Sousa
Departamento de Ciência dos Computadores
Faculdade de Ciências da Universidade do Porto
Porto, Portugal
up201504217@edu.fc.up.pt

***Abstract*—O *Mobile IP* (ou *MIP*) foi projetado para ser um protocolo de comunicação padrão da *IETF* de forma a atender às necessidades do crescente número de usuários de dispositivos móveis que pretendam para além da conetividade manter também as suas comunicações adaptadas à sua mobilidade no dia a dia. Neste trabalho iremos começar com uma breve descrição deste protocolo, com alguns detalhes que se destacam no mesmo, irá ser feita também uma demonstração com o intuito de mostrar o funcionamento do mesmo, nas duas versões que existem atualmente (*MIPv4* e *MIPv6*).**

Keywords—Mobile IP(MIP), mobility, MIPv4, MIPv6

I. Introdução

Nos últimos anos, a utilização de dispositivos móveis aumentou exponencialmente[2] mas as versões atuais do protocolo de Internet não suportam a mobilidade do *host*. Estas foram projetadas para que o ponto de conexão de um nó à rede permanece inalterado o tempo todo e para que um endereço IP identifique a rede em que se encontra.

Assim sendo, foi necessário fazer uma reconfiguração de forma a criar um suporte para estes *hosts* móveis o que implicava modificar o endereço *IP*, ter um novo prefixo, mas seria necessário deixar as conexões *TCP* (usados pelos serviços de Internet), atualizações de *DNS* (levando a atrasos de propagação) e até a problemas de segurança.

Sseria também necessário alterar as tabelas de roteamento para entregar pacotes para um novo local, o que não seria escalável (oscilações na rede e a um aumento de terminais móveis) e mais uma vez problemas de segurança. Foram estas as motivações que levaram à ascensão do *Mobile IP*[3].

II. Descrição da Tecnologia

Antes de iniciar a descrição do funcionamento da parte prática do trabalho com mais detalhe, vamos fazer uma pequena introdução ao tema, começando por falar do *Mobile IP* bem como a sua utilização nas duas versões do protocolo *IP*, o *IPv4* e o *IPv6*.

A. O que é o Mobile IP?

Mobile IP é um protocolo de comunicação através do qual os utilizadores se podem mover entre redes diferentes mantendo um endereço *IP* permanente (*home address*) proveniente da sua rede de origem, isto com a ajuda dos agentes de mobilidade que existem neste protocolo, vai ajudar a que seja possível comunicar com estes utilizadores normalmente.

B. Terminologias

Para facilitar a compreensão do relatório indicamos abaixo as terminologias mais usadas.

- *Mobile Node (MN)* – o dispositivo do utilizador que vai andar móvel, este tem um endereço IP permanente que lhe é atribuído dentro do *home network* chamado *home address*.
- *Home Agent (HA)* – dispositivo que está na *home network*, normalmente um *router* mas não necessariamente, e está conectado ao *mobile node* quando este está na *home network*.
- *Home Network* – rede de onde o *mobile node* é originalmente.
- *Foreign Agent (FA)* – dispositivo que vai ser o *default router* do *mobile node* na *foreign network*, pode não existir se o *mobile node* optar por ter um *care-of address* localizado nele próprio em vez de ser num dispositivo à parte.
- *Foreign Network* – rede visitada pelo *mobile node*.
- *Correspondent Node (CN)* – Nó que vai querer comunicar com o *mobile node*.
- Care-of Address (CoA) – endereço IP que está numa das extremidades do túnel entre os agentes de mobilidade, este vai representar o local (rede) onde o *mobile node* está naquele momento, este endereço pode ser partilhado por vários dispositivos da rede visitada que tenham o mesmo *foreign agent*.

C. Como funciona o MIP?

O funcionamento deste protocolo trata várias partes ligadas à mobilidade de dispositivos pelas redes e cada uma dessas partes é importante para que este protocolo funcione corretamente. Abaixo vai ser feita uma explicação de como acontecem coisas como: *MN* saber se está na sua *home network* ou numa *foreign network*, como acontece na atribuição de um *CoA* e a forma como essa informação chega ao *HA* e também como ocorre o envio de pacotes do *CN* para o *MN*.

1) MN desloca-se para uma nova rede

Os agentes de mobilidade, o *HA* e *FA*, vão enviando *broadcast agent advertisements* que o *MN* recebe e é desta forma que ele percebe se está na *home network* ou numa foreign network. Também é possível que o *MN* envie *agent solicitation* em vez de só ficar à espera de receber os *agent advertisements* dos agentes de mobilidade.

Ao chegar a uma *foreign network* o *MN* vai receber de um *FA* um *agent advertisement* e vai perceber que já não se encontra na sua *home network*, quando isto acontece o *MN* vai responder ao *advertisement do FA* e este vai tornar-se o *router*

default do *MN* e vai associar-lhe um *CoA* enquanto ele estiver nessa rede.

Após a atribuição de um *CoA* o *MN* vai fazer o registo com o *HA*. O registo vai ser feito da seguinte forma: o *MN* vai enviar o pedido de registo (*registration request*) para o *FA* e o *FA* vai enviá-lo para o *HA*, desta forma o *HA* vai ficar a saber qual a localização do *MN* e para onde enviar os pacotes que lhe estiverem destinados, depois de receber o pedido de registo o *HA* vai responder a esse pedido com um *registration reply* que é enviado para o *FA* e deste para o *MN* ficando efetuado o registo nesse momento. Este registo vai ter um tempo de vida e vai ter de ser renovado sempre que este tempo estiver a chegar ao fim. No caso do registo não ser renovado antes do tempo chegar ao fim vai acontecer a remoção do mesmo.

2) CN envia pacotes para o MN

O *CN* vai querer enviar pacotes ao *MN* mas este já não está na *home network*, como é que o *CN* vai conseguir fazer que os pacotes cheguem ao seu destino sendo que não pode enviar diretamente para o *MN* pois não sabe onde este está?

Os pacotes vão ser enviados do *CN* com o endereço IP do mesmo como endereço de origem e com o endereço IP permanente do *MN*, ou seja, o endereço IP que o *MN* tem da *home network*. No entanto quem vai receber esses pacotes vai ser o *HA*, ao ver que esses pacotes são para o *MN* ele vai indicar ao *CN* que pode enviar para ele próprio porque o *HA* ficou com o endereço IP do *MN* da *home network* enquanto ele não está na mesma.

Quando recebe os pacotes o *HA* vai encaminhá-los para o *MN* sendo que para isso envia o que recebeu para o *FA* por um túnel que os conecta. Dentro deste túnel os pacotes vão ser enviados utilizando um mecanismo que tem como nome encapsulação[5] a isto é dado o nome de *tunneling*, o *HA* vai encapsular o pacote que recebeu para o *MN* e pôr como endereço IP de origem o seu endereço e o endereço de destino será o *CoA* para enviar o pacote para o *FA*.

Depois de receber os pacotes do *HA*, através do túnel, o *FA* vai retirar a encapsulação e enviá-los para o *MN*. Após receber os pacotes o *MN* vai mandar uma resposta para o *FA* e este vai reencaminhar diretamente para o *CN*.

3) No caso do MN ter o CoA localizado nele próprio

Pode acontecer que o *MN*, em vez de escolher um router na rede visitada para ser o seu *router default*, adquire um *CoA* que vai ficar localizado no *MN* em si. Neste caso o *MN* vai adquirir um *CoA* por *Dynamic Host Configuration Protocol(DHCP)* ou *Point-to-Point Protocol(PPP)* e em vez de o túnel ser entre agentes de mobilidade vai ser entre o *HA* e o *MN*, sendo que o extremo do lado do *MN* vai ter como endereço IP o *CoA*.

4) Proxy ARP e Gratuitous ARP

Neste protocolo existem dois tipos de mensagens *ARP* que são usadas, *proxy ARP* e *gratuitous ARP*.

As mensagens *Proxy ARP* são mensagens enviadas pelo *HA*, em nome do *MN*, e que ajudam os outros nós da *home network* a comunicarem com o *MN* quando este está fora. No caso das mensagens de *gratuitous ARP*, estas são mensagens enviadas para fazer *updates* às *caches* normalmente usadas pelo *HA* quando há movimentação do *MN* para outra rede ou quando este volta à *home network*.

O *MN* não deve enviar mensagens *ARP*, *proxy ARP* ou *gratuitous ARP*, a partir do seu *home address* se estiver fora da sua *home network*, mensagens de *ARP* só devem ser trocadas com o *FA* nesse momento.

D. MIPv4

Como o IPv4 foi implementado ainda antes de haver necessidade de mobilidade foi necessário fazer a adaptação para que funcionasse com Mobile IP. Nesta versão o *MIP* funciona como descrito acima.

No entanto esta forma de tratar a mobilidade de um dispositivo não é a mais eficiente e pode levar a perdas de pacotes. Nesta versão conseguiu corrigir-se o último ponto mas continua a não ser a forma mais eficiente, no entanto foi a forma encontrada de maneira a que funcionasse em *IPv4* com as ferramentas que este tinha disponíveis.

O problema que pode existir com a forma de funcionar do *MIP* descrita acima tinha haver com o facto de alguns *routers* fazerem a verificação da topologia do endereço IP através do qual são enviados os pacotes. No caso de isto acontecer quando um *router* recebesse um pacote vindo do *MN* tendo como endereço de origem o endereço permanente deste na *home network* e comparasse esse endereço com o prefixo da rede visitada (rede de onde veio o pacote) iria ver que o endereço não está topologicamente correto e ia deitar fora o pacote recebido.

A solução encontrada foi a possibilidade de criação de um túnel no sentido oposto ao túnel que foi referido anteriormente. Desta forma o *MN* durante a fase de registo pode fazer o pedido para que seja criado um túnel no sentido inverso. Este novo túnel vai ter no seu início o *CoA* do *MN* e o seu final vai ser o endereço do *HA*. Isto é chamado de *reverse tunneling*, assim em vez de o *MN* enviar a resposta para o *FA* e este encaminhar diretamente para o *CN*, o que vai acontecer é que o *FA* vai enviar a repostas para o *HA* e este envia-as para o CN. Assim quando um *router* fizer a verificação da topologia esta já vai estar correta o que faz com que o pacote não seja perdido mas sim enviado para o seu destinatário.

E. MIPv6?

Quando *IPv6* foi criado já foi tido em conta as coisas que seriam necessárias para que funcionasse com *MIP* e também formas de facilitar o uso deste protocolo e torná-lo mais eficiente do que na versão *MIPv4*, como a adição de um *mobility header* para as mensagens de mobilidade.

No *MIPv6* foi encontrada uma forma de não só evitar a perda de pacotes por topologia incorreta do endereço *IP* de origem, mas também de tornar a troca de mensagens mais eficiente e segura entre o *MN* e o *CN*.

Nesta versão vai acontecer uma primeira fase em que o *CN* passe vai procurar saber onde se encontra o *MN* e como o contactar diretamente, mas após esta fase o contacto entre estes dois dispositivos já vai ser direto sem necessidade de passar alguma coisa pelo *HA*.

1) CN tenta comunicar com MN pela primeira vez

O *CN* numa primeira tentativa de contactar o *MN*, depois deste último estar a visitar outra rede, não sabe onde ele se encontra. Então o *CN* vai enviar os pacotes que tem para o *MN* com o endereço de origem sendo o endereço *IP* dele próprio e o endereço de destino vai ser o home address do *MN*, como o

MN já não se encontra na home network, os pacotes vão ser recebidos pelo *HA* que os vai mandar pelo túnel que o liga ao *FA* para desta forma os pacotes chegarem ao *MN* que está numa *foreign network*.

Quando os pacotes chegam ao *MN* este vai enviar um *binding update* ao *CN* para que este último saiba que sempre que quiser comunicar com ele pode fazê-lo diretamente através do seu *CoA*. Após receber este binding update o *CN* vai guardar a associação do *home address* ao *CoA* do qual recebeu o *binding update* numa entrada na sua *cache* e vai enviar ao *MN* um *binding acknowledgment* para informar que já guardou informação, assim o *MN* já pode usar o seu *CoA* como endereço de origem dos pacotes sem haver desconfiança da parte do *CN*.

Nesta versão de *MIP* o *CN* vai ter uma *binding cache* onde vai guardar os *CoAs* de todos os *MNs* com os quais consegue comunicar. Esta *binding cache* vai ser atualizada sempre que o *CN* receba um *binding update* de um *MN* por isso estas mensagens vão sendo enviadas regularmente para fazer *refresh* ao tempo de vida da associação entre os dois endereços. Esta *binding cache* também vai existir no *HA* funcionando da mesma forma que a do *CN*.

2) Restantes trocas de mensagens entre CN e MN

Após o *CN* ter a associação do *home address* do MN com o *CoA* do mesmo a comunicação entre estes dois nós vai ser direta e sem necessidade de intermediários, que seriam os agentes de mobilidade neste caso, enquanto a *cache entrie* tenha tempo de vida maior que 0. Estas entradas na tabela de *cache* vão sendo atualizadas com os *binding updates* enviados pelo *MN*, quando o tempo de vida de uma entrada chega a 0 o *CN* deixa de associar o *home address* a um *CoA* e volta a fazer o envio de pacotes para o *home address*.

3) MN volta à home network

Ao voltar à home network o *MN* vai voltar a usar o *home address*, no entanto esse endereço tem sido utilizado pelo *HA* para receber os pacotes que eram para o *MN* enquanto este esteve fora. Para "recuperar" o seu *home address*, e informar o *HA* que o voltou a usar como endereço de origem, o *MN* vai ter de descobrir qual é o endereço *IP* do *HA* dentro da sua *home network* enviando uma mensagem de *neighbour solicitation*.

Então o *MN* vai enviar a mensagem de *neighbour solicitation* onde vai deixar o campo de endereço de destino como não especificado (::) e vai ter no campo de destino um solicited-node multicast address sendo que o objetivo vai ser o *home address*. O *solicited-node multicast address* vai ser utilizado para descobrir se o *home address* já está a ser usado com endereço *IP* de algum nó na rede, isto acontece com a ajuda do processo chamado *Duplication Address Detection (DAD)*.

A esta mensagem o *HA* vai responder com um *neighbour advertisement* para avisar que esse endereço já está a ser usado (pelo *MN*) e é ao receber esta mensagem que o *MN* vai descobrir o endereço do *HA* para lhe puder enviar um *binding update*. Com este *binding update* o *HA* vai perceber que o *MN* voltou para a *home network* e assim limpa o *CoA* que estava associado ao *home address*.

4) Autenticidade dos binding updates

A autenticidade dos *binding updates* enviados pelo *MN* vai ser assegurada por um processo chamado de *return routability*.

Através deste processo o *MN* vai provar ao *CN* que ao receber um *binding update* seu pode comunicar com ele tanto utilizando o *home address* como utilizando o *CoA*. Mas não é possível ter uma associação de segurança com todos os *CNs* que possam querer comunicar com o *MN*, portanto é utilizado o processo de *return routability* para conseguir daí retirar uma chave que será usada para manter segura a comunicação entre os dois nós.

Ao mesmo tempo também vai fazer com que o *HA* só receba *binding updates* para um *home address* do *MN* em questão, ou seja um *MN* não pode enviar *binding updates* em nome de outro *MN*. Neste caso é usado o protocolo *IPsec* entre o *HA* e o *MN* para criar uma associação de segurança e assim manter a comunicação segura.

III. Objetivos do projeto

Neste trabalho, o objetivo principal é a implementação de duas redes com suporte *MIP*, usando um simulador de redes como recurso central das mesmas, que serão feitas para *IPv4* e *IPv6* por forma a estudar as diferenças entre os dois protocolos a nível de mobilidade.

Para tal iremos recorrer a ferramentas e componentes para construir este sistema, começando com o *GNS3* que será a ferramenta central, e o *mipv6-daemon* e o *radvd* como componentes auxiliares na construção da rede. Ao longo da criação da parte prática pretendemos implementar todas as caraterísticas de cada versão do *MIP*, explorando diversas funcionalidades das mesmas de forma a poder compará-las.

IV. Identificação de ferramentas/componentes a usar

A. Que ferramenta vamos usar para o trabalho?

A ferramenta principal será o *GNS3* que será instalado num *VM* com sistema operacional *fedora* e onde serão criadas as redes.

Posteriormente outros componentes serão adicionados por forma a conseguir implementar todas das funcionalidades das duas versões *MIP*.

1) GNS3

O GNS3 é um simulador de redes completas, que permite simular, configurar, testar e solucionar problemas de redes virtuais e reais com recurso a diversos equipamentos ativos de uma rede como *routers, switchs, PCs,* telefones, *firewalls*, entre outros.

Como se trata de um aplicativo de fácil utilização e que já estamos familiarizados decidimos optar por esta opção uma vez que se enquadrava na nossa ideia de criar redes virtuais para demonstrar o funcionamento do *MIP* em ambas as versões.

2) mipv6-daemon

O daemon IPv6 móvel permite que os nós permaneçam acessíveis enquanto se movimenta na Internet IPv6. Isto é possível através de pacotes, que contêm um serviço IPv6 móvel, para clientes permitindo que eles sejam realocados em uma rede habilitada para IPv6 e ainda assim sejam alcançáveis.

3) radvd

O *router advertisement daemon* (*radvd*) será executado como router *IPv6* e será responsável por enviar mensagens de anúncio de router, para a *LAN Ethernet local* periodicamente,

quando solicitado por um nó que envia uma mensagem de *router solicitation*.

B. *Demonstração de viabilidade*

Como já referimos anteriormente, a ideia será a de criar duas redes com suporte *MIP*.

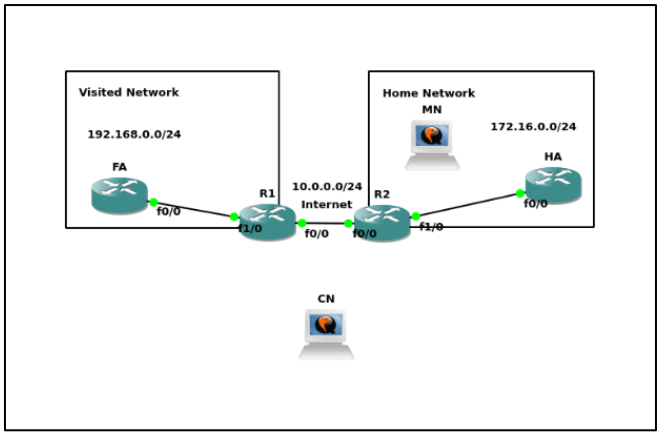

Fig. 1. Exemplo de uma rede MIP simples

Descrição da Figura 1: Na figura temos duas *networks*, uma designada *Home Network* onde se encontra o *mobile node*(*MN*) e o *home agent*(*HA*), neste caso um *router*, e ainda contém um *router*(*R2*) com uma interface nesta rede e que será o intermediário de ligação da rede *home* à rede visitada.

Já na outra rede, a *Visited Network*, temos um *router* que é o *foreign agent*(FA) e outro *router*(*R1*) que terá a mesma finalidade do *R2*.

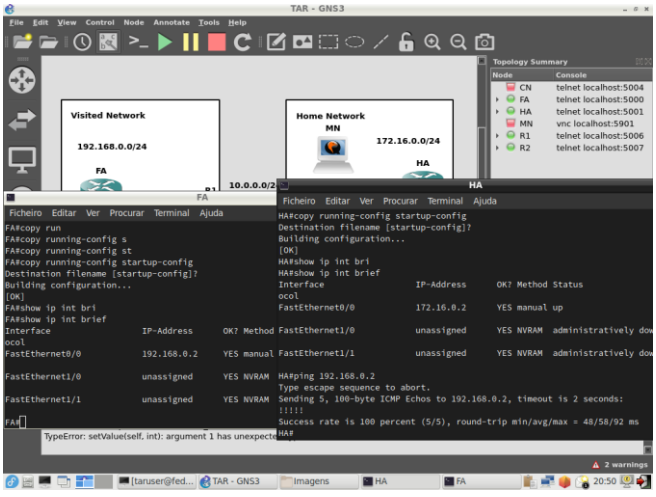

Fig. 2. Comunicação entre HA e FA

Descrição da Figura 2: Na figura podemos ver que mesmo em redes diferentes o *home agent* consegue comunicar com o *foreign agent*.

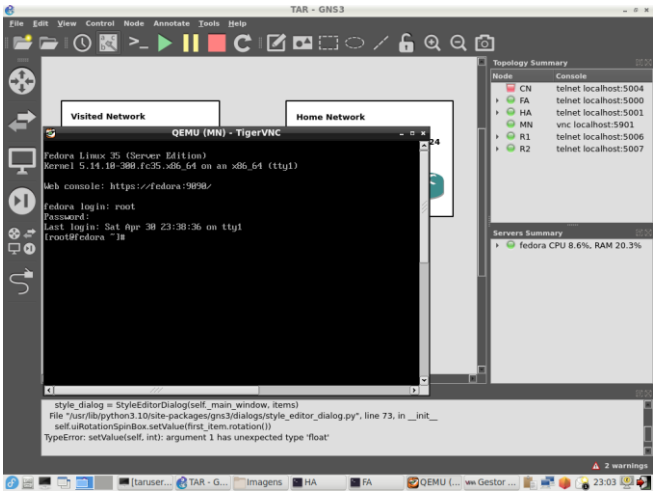

Fig. 3. Exemplo do mobile node

Descrição da Figura 3: Na figura podemos ver que o mobile node(MN) é um terminal linux(fedora server).

Na primeira rede, com suporte para *IPv4*, será usado um terminal, que simulará um terminal *linux*, e que funcionará como *MN*, e que está conectado à sua rede doméstica (*home network*). Caso este pretenda visitar uma rede, terá de se conectar pedindo ao *FA* (neste caso um *router*) um *CoA* através de uma rota *ICMP* propagada. Em seguida, uma mensagem de registo é enviada pelo *MN* para o *HA* (também um *router*) usando o *FA* como nó intermediário.

Esta mensagem permite que o *MN* atualize o seu *HA* com o novo *CoA*. Após este receber esta mensagem, ele cria uma entrada local, mapeando o endereço IP da *home* do *MN* com o *CoA*. Para concluir este processo de registo, é enviada uma confirmação de resposta em formato mensagem, do *HA* para o *FA*. Então, o *FA* reenvia esta última mensagem para o *MN*, concluindo assim o processo de registo.

Na rede com suporte para *IPv6* a execução será semelhante, mas fazendo as adaptações necessárias como o *HA* e o *FA* deixarem de ser intermediários na troca de mensagens, passando assim a comunicação a ser feita entre *MN* e *CN* dos dois modos conhecidos, *Tunneling* e Otimização de rota.

V. Planeamento do resto do trabalho

Conforme já referimos nos pontos anteriores, a ideia será a criação de duas redes MIP, uma com suporte para IPv4 e outra para IPv6.

Nesta fase inicial e para mera demonstração, implementamos apenas uma rede simples de MIP mas depois será necessário modificá-la para funcionar como MIPv4, criar a outra rede, a de MIPv6, e posteriormente implementar o *triangular routing* e *reverse* tunneling entre outras configurações que permitam simular o funcionamento de cada rede e fraquezas de cada versão do MIP.

Por fim iremos estudar a diferenças entre as duas versões do protocolo, quer a nível de eficiência como de desempenho, usando o *WireShark* no *GNS3*, por exemplo, para ir observando e analisando o tráfego de rede, monitorizando assim a entrada e saída de dados.

REFERENCES

Nesta parte iremos incluir todas fontes em que nos baseamos para escrever este relatório.

[1] Rui Prior, slides das aulas de Tópicos avançados em redes que foram baseados no livro de Hesham Soliman "Mobile IPv6: Mobility In A Wireless Internet", Addison-Wessley, 2004, último acesso a 30 de abril 2022

Nicolas Zwierzykowski, "A evolução dos dispositivos móveis e a sua influência em nossas vidas", https://pt.linkedin.com/pulse/evolu%C3%A7%C3%A3o-dos-dispositivos-m%C3%B3veis-e-sua-influ%C3%ANcia-em-zwierzykowski, último acesso a 30 de abril de 2022

[2] Charles M. Kozierok , "Mobile IP Overview, History and Motivation", http://www.tcpipguide.com/free/t_MobileIPOverviewHistoryandMotivation-3.htm, último acesso a 30 de abril de 2022

[3] Yi-an Chen, "A Survey Paper on Mobile IP", https://www.cse.wustl.edu/~jain/cis788-95/ftp/mobile_ip/index.html, último acesso a 30 de abril de 2022

[4] Charles E. Perkins, Sun Microsystem, "Mobile IP", http://wmnlab.ee.ntu.edu.tw/951cross/MobileIP_CommMag1997.pdf, último acesso a 30 de abril de 2022

[5] Raman Bhadauria, "Mobile Internet Protocol (or Mobile IP)", https://www.geeksforgeeks.org/mobile-internet-protocol-or-mobile-ip/, último acesso a 30 de abril de 2022

[6] Oracle Corporation, "How Mobile IP Works", https://docs.oracle.com/cd/E19455-01/806-7600/6jgfbep13/index.html, últmio acesso a 30 de abril de 2022

[7] Damien Phillips, RMIT University, Jiankun Hu, UNSW Sydney, "Simulation Study of TCP Performance Over Mobile IPV4 and Mobile IPV6", https://www.researchgate.net/publication/220710456_Simulation_Study_of_TCP_Performance_Over_Mobile_IPV4_and_Mobile_IPV6, último acesso a 30 de abril de 2022

[8] Autor desconhecido, "Solicited-node multicast address", https://en.wikipedia.org/wiki/Solicited-node_multicast_address, último acesso a 30 de abril de 2022

[9] GNS3, "Getting Started with GNS3", https://docs.gns3.com/docs/, último acesso a 30 de abril de 2022

[10] Fabrice Bellet, "mipv6-daemon-1.0-13.fc31 RPM for aarch64", http://rpmfind.net/linux/RPM/fedora/32/aarch64/m/mipv6-daemon-1.0-13.fc31.aarch64.html, último acesso a 30 de abril de 2022

[11] Radvd team, "Linux IPv6 Router Advertisement Daemon (radvd)", https://radvd.litech.org/, último acesso a 30 de janeiro de 2022

[12] Nautilus Project, "About Nautilus6", https://www.nautilus6.org/, último acesso a 30 de abril de 2022