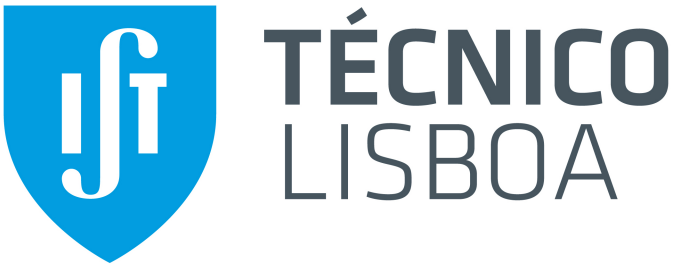

Instituto Superior Técnico

Mestrado Integrado Em Engenharia Eletrotécnica e de Computadores

Redes Móveis e Sem Fios

Lab 2

Trabalho realizado por:	*Número:*
Diogo Moura	86976
Diogo Alves	86980
Luís Crespo	87057

Turno L03 - Grupo 1 - 5a feira 14h:00m-15h30m

2019/2020

Conteúdo

1 Introdução

Este trabalho destina-se a estudar os processos e protocolos envolvidos nas ligações *ad hoc* e a identificar e solucionar os problemas que surgem nessas mesmas redes, por exemplo, quando um nó da rede se movimenta.

2 Work Description

2.1 Configuring an IPv4 Network

Ver os ficheiros contidos em anexo.

2.2 IEEE 802.11 Access Network

2.2.1 How many BSSs and ESSs are there in this scenario?

Neste cenário existem dois BSS (*Basic Service Set*). Cada um é constituído pelo respetivo *access point* e identificado pelo seu SSID. O primeiro BSS utiliza o *accessPoint1* e o seu SSID é *eduroam* e o segundo BSS utiliza o *access point 2* e tem como SSID *INESC-ID*.

Estes dois BSS pertencem a um único ESS (*Extended Service Set*). O ESS inclui também o *distribution system* (DS), que neste caso corresponde à rede de *routers* e computadores com fios à qual os *access points* estão ligados. O ESS permite que os *hosts* sem fios circulem entre os dois BSS sem que eles percam a conexão.

2.2.2 In order to be able to communicate through the WLAN, the stations must engage in a number of management procedures: Scanning, Authentication, Association. Calculate how many data packets could be transmitted during the interval between the start of Scanning and the end of Association.

O intervalo entre o início do *scanning* e o fim da *association*, isto é, o intervalo entre o primeiro *Probe-Request* do *Host* para o AP e o *frame* "AssocResp-OK"do AP para o *Host* é de 1.566128-0.263503=1.3011s. Cada pacote de dados tem 100B de *payload* mais 71B de *overhead*, perfazendo um total de 171B e é transmitido a um ritmo de 24Mbps. Admitindo que o tempo de transmissão e espera nos *routers* é desprezável face ao tempo de transmissão na rede sem fios, então seria possível transmitir $1.3011/\frac{171*8}{24*10^6} = 22826$ pacotes neste intervalo de tempo.

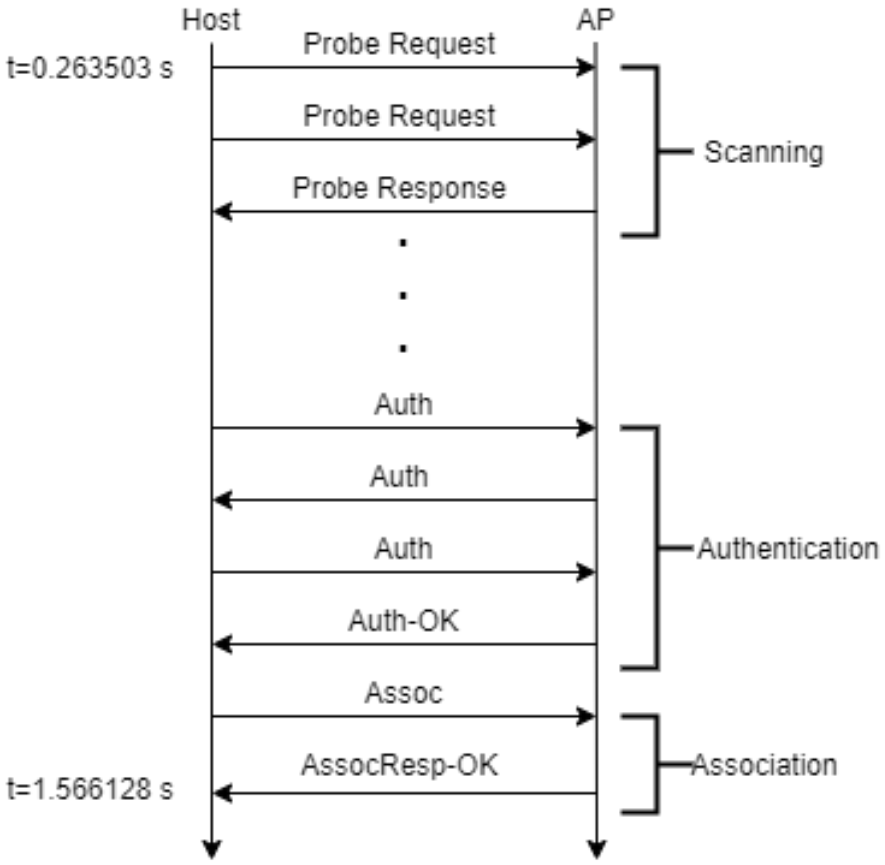

Figura 1: *Diagrama temporal dos pacotes trocados durante o processo de associação*

2.2.3 Why does the first ProbeReq get no answer from accessPoint1, while the second ProbeReq gets an answer?

O primeiro *frame "Probe Request"* enviado pelo *wireless Host* não obtém qualquer resposta de nenhum *access point*. Já o segundo *frame* obtém um resposta *"Probe Response"* do *accessPoint1*.

Observando o módulo *mgmt* da interface wireless *wlan[0]* do *wirelessHost1*, verifica-se que o atributo *"scanning"* tem o campo *curChan* igual a 0 quando é enviado o primeiro *"Probe Request"* e igual a 1 quando é enviado o segundo. De facto, na fase de *"scanning"*, o *wirelessHost* envia um *probe Request* em cada canal (0, 1, 2, 3, 4) até obter uma resposta. Como cada canal corresponde a uma gama de frequências de operação diferente, e cada *access point* apenas opera num canal, o *frame "Probe Response"* só será enviado pelos *access points* que operarem no mesmo canal em que está o *"Probe Request"* enviado.

Neste caso, o *accessPoint1* opera no canal 1 e o *accessPoint2* opera no canal 3, o que explica o facto de não haver resposta ao primeiro *Probe* (canal 0), mas existir resposta ao segundo (canal 1).

2.3 IP Networking and Mobility Issues

2.3.1 The IEEE 802.11 data packet sent by wirelessHost1 towards wiredHost1 or wiredHost2 use three address fields of the IEEE 802.11 header. One of these fields contains the MAC address of router1. How does wirelessHost1 know that IEEE 802.11 frames containing IP packets sent towards wiredHost1 or wiredHost2 should be sent to router1?

No pacote de dados enviados do *wirelessHost1* para o *wiredHost1/wiredHost2*, o campo *"address3"* do *header* IEEE802.11 tem o endereço MAC do *router1*. Para determinar o significado deste campo, olha-se para o campo "toDS" e "fromDS" deste *header*. Neste caso, "toDS" = TRUE e "fromDS" = FALSE, o que significa que o campo *"address3"* possui o *"Destination Address"*. Assim surge a pergunta: como é que o *wirelessHost* sabe que os pacotes com destino ao *wiredHost1/wiredHost2* devem ser enviados através do *router1*?

Para responder a este questão é necessário analisar os pacotes trocados antes da primeira tentativa de envio de um pacote do *wirelessHost1* para um dos *wired hosts*. Verificou-se que, de modo a determinar o endereço MAC correspondente ao endereço IP do *wiredHost1/2*, o *wirellesHost* envia um pacote *"ARPRequest"* em que o campo *"destIP"* contém o endereço IP do *wiredHost1/2*. Como resposta a esta mensagem, recebe um pacote *"ARPReply"* em que o campo *"srcIP"* é de facto o endereço IP do *wiredHost1/2*, mas o campo *"srcMAC"* corresponde ao endereço MAC do *router1*. De facto, este pacote *"ARPReply"* teve origem no *router1*, uma vez que os pacotes com destino ao *wiredHost1/2* serão reencaminhados através deste *router*, pelo que se um pacote da rede 10.0.1.0/24 tiver como IP de destino o endereço IP do *wiredHost1/2*, então deverá ter como *"Destination Address"* (campo *"address3"* do *header* MAC) o endereço MAC do *router1*.

Assim, ao receber o *"ARPReply"*, o *wirelessHost1* atualiza a sua tabela ARP de forma a associar o endereço IP do *wiredHost1/2* ao endereço MAC do *router1*.

2.3.2 Simulate the scenario during 400 s. Compare the total number of data packets sent by the applications of wiredHost1 with the total number of packets received by the same applications. Explain the results

Ao correr o cenário durante 400s e observar os resultados, verifica-se que os *wiredHost1* e *wiredHost2* enviam cada um 400 pacotes, mas só recebem 187. Já o *wirelessHost1* recebe 374 e envia também 374 (correspondestes ao *echo* dos pacotes recebidos). Assim, podemos concluir que todos os pacotes enviados pelo *wirelessHost1* são recebidos pelos *wiredHost1* e *wiredHost2*, uma vez que 187+187=374, mas nem todos os pacotes enviados pelos *wiredHost1/2* são recebidos pelo *wirelessHost1*. Observando o diagrama temporal

dos pacotes recebidos pelo *wirelessHost1* (figura 2), verifica-se que os pacotes apenas são recebidos no inicio e no fim da simulação, que corresponde ao período em que este se encontra associado ao *accessPoint1*. Quando o *wirelessHost1* está associado ao *accessPoint2*, não existem pacotes recebidos. De facto, quando o *wirelessHost1* se associa ao *accessPoint2*, este deixa de receber pacotes do *accessPoint1*. Como os *wiredHosts* enviam sempre os pacotes para o mesmo endereço IP (*10.0.1.2*, que é o endereço de IP fixo do *wirelessHost1*), eles vão sempre inevitavelmente parar ao *accessPoint1*, uma vez que as tabelas de encaminhamento dos routers também são fixas.

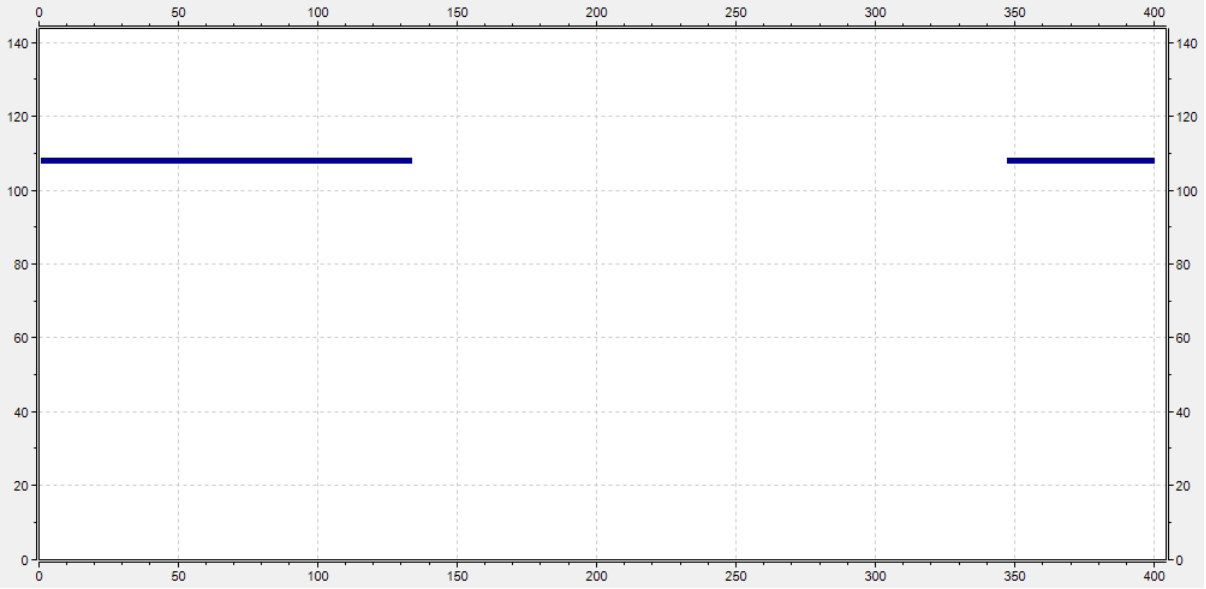

Figura 2: *Diagrama temporal dos pacotes recebidos pelo wirelessHost1*

2.3.3 How would you solve the problems identified in Q.2.3.2.?

Para resolver este problema teremos que utilizar *mobile IP tunneling*.

Este processo é constituído por 3 fases:

- *Agent Discovery* - o nó descobre os seus *foreign agent* e *home agent*.
- *Registration* - o nó regista a sua localização atual nos *foreign agent* e *home agent*
- *Tunneling* - um túnel bi-direcional é estabelecido pelo *home agent* para o *care of address* (nova localização do nó na rede remota), para fazer *routing* dos pacotes para o nó móvel, enquanto o mesmo se desloca

Agent Discovery

Durante a fase *Agent Discovery*, o *home agent* e o *foreign agent* anunciam os seus serviços à rede, utilizando o *ICMP Router Discovery Protocol* (IRDP). O nó móvel escuta estes anúncios para determinar se está ligado à sua rede de origem (*home network*) ou a outra (*foreign network*).

Se um nó móvel determinar que está ligado a uma *foreign network*, tenta adquirir um *care of address*.

Registration

O nó móvel, o *foreign agent* e o *home agent* trocam mensagens e autenticam-se para estabelecer uma ligação de *tunneling*. O nó móvel pode registar-se novamente antes que o tempo de vida do seu registo expire.

O *foreign agent* adiciona o nó móvel à sua lista de visitantes e estabelece um túnel para o *home agent*, criando uma entrada de *routing* para entregar os pacotes ao *home address*.

Tunneling

O *tunneling* tem duas funções principais: encapsulamento dos pacotes de dados para chegar ao *endpoint* do túnel e desencapsulamento dos pacotes quando são entregues ao *endpoint*. O nó móvel envia os pacotes utilizando o seu *home IP address*, mantendo a aparência de que permanece na sua rede, mesmo quando se desloca para *foreign networks*.

Pacotes de dados endereçados para o nó móvel são encaminhados para a sua *home network*, onde o *Home agent* interceta e faz *tunneling* dos mesmos para o *care-of address*, destinado ao nó móvel, como representado na figura 3.

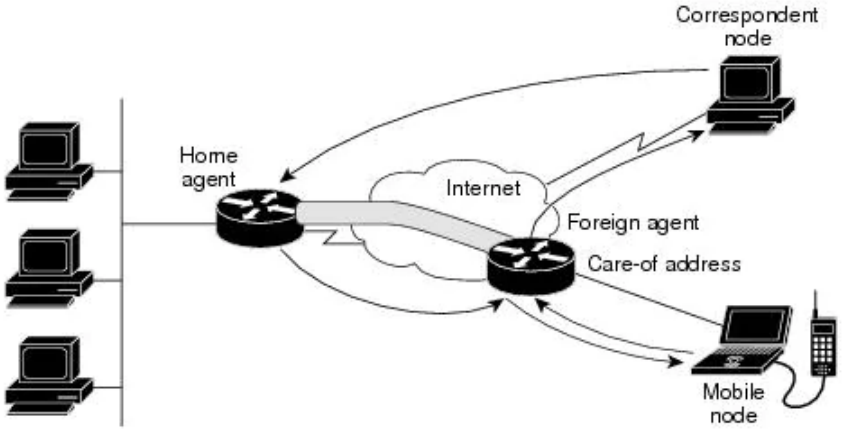

Figura 3: *Packet Forwarding*

Quando o nó móvel envia pacotes o *foreign agent* faz *tunneling* dos mesmos. Como os pacotes do nó móvel mostram a *home network* como fonte dentro da *foreign network*, uma *access control list* nos *routers* chamada de *ingress filtering* descarta os pacotes em vez de os encaminhar. Uma funcionalidade chamada de *Reverse Tunneling* resolve este problema, fazendo com que o *foreign agent* faça *tunneling* dos pacotes para o *home agent* quando os recebe do nó móvel, como está representado na figura 4.

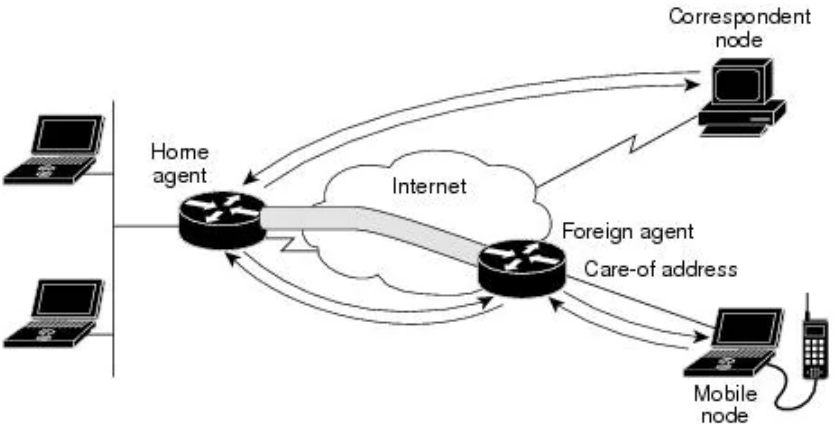

Figura 4: *Reverse Tunneling*

Mapeando estas entidades para o nosso caso, o *Mobile Node* será o *wirelessHost1*, o *Home Agent* será o *accessPoint1* e a *Home Network* a rede com SSID *eduroam*. O *Foreign Agent* será o *accessPoint2* e a *Foreign Network* será a rede *INESC-ID*. O *Care of Address* será o novo endereço do *wirelessHost1* nesta rede (que poderá ser obtido por DHCP). Por fim, os *Correspondent Nodes* seráo os *wiredHost1* e *wiredHost2*.