# Trabalho Prático Nº3 – Nível de Ligação Lógica: Redes Ethernet e Protocolo ARP

Duração: 4h

Neste trabalho deve usar a máquina nativa e máquina virtual XubunCORE_7_5 (TP0) para a última questão.

Nota importante: O trabalho é para ser realizado nas aulas PL correspondentes. Não serão aceites trabalhos "resolvidos em casa".

## 1. Objetivo

O objetivo deste trabalho é explorar a camada de ligação lógica, focando o uso da tecnologia Ethernet e o protocolo ARP (*Address Resolution Protocol*).

O protocolo ARP (ver RFC 826 (http://tools.ietf.org/html/rfc826.html) é usado pelos equipamentos em rede para efetuar o mapeamento entre os endereços de rede e os endereços de uma tecnologia de ligação de dados, vulgarmente designados por endereços MAC (*Medium Access Control*). Desta forma, o protocolo ARP permite determinar, por exemplo, qual o endereço Ethernet que corresponde a um endereço IP particular.

## 2. Introdução

Um dos conceitos mais importantes de uma pilha protocolar estruturada em níveis ou camadas é que cada camada fornece serviços às camadas superiores e usa os serviços disponibilizados pelas camadas inferiores. Por exemplo, a camada de ligação lógica oferece os seus serviços à camada de rede e através dela às camadas superiores (transporte e aplicação) e utiliza, por sua vez, os serviços da camada física.

O serviço básico prestado pela camada de ligação lógica é a **transferência de dados** de um nó para os nós imediatamente adjacentes na topologia da rede. Em cada nó de origem, a unidade protocolar de dados (*Protocol Data Unit* (PDU)) do nível de rede (datagrama IP) e é encapsulado num PDU do nível de ligação (trama), sendo depois enviado através da camada física para o nó adjacente. Por sua vez, este nó recebe a trama do nível físico, extrai o datagrama IP da trama recebida e entrega-o ao nível de rede para ser processado.

Outros serviços que um protocolo do nível de ligação lógica pode fornecer são: controlo de acesso ao meio, entrega fiável de dados, controlo de fluxo e controlo de erros (deteção e correção), embora não estejam disponíveis em todos as tecnologias do nível da ligação de dados (nível 2). Estes serviços podem ser oferecidos por outros níveis da pilha protocolar, por exemplo, o nível de transporte com o protocolo TCP *(Transmission Control Protocol*). A principal diferença é que no nível de ligação estes serviços são prestados na ligação entre nós adjacentes enquanto no nível de transporte são prestados fim-a-fim. Neste caso, uma ligação fim-a-fim envolve normalmente a travessia de um percurso na rede que passa por múltiplos nós intermédios.

### Deteção e Correção de Erros

A deteção e correção de erros é outro exemplo de uma funcionalidade de serviço que pode ser prestada nos vários níveis da pilha protocolar.

Genericamente a deteção e correção de erros ao nível de ligação lógica, bastante mais sofisticada que nos níveis protocolares superiores, consegue detetar (e eventualmente corrigir) erros de um bit e alguns erros com vários bits. O mecanismo de deteção mais comum é baseado num bloco de bits (B) criado pelo originador, que é uma função *f* da informação presente na trama a ser transmitida. Esse bloco de bits é acrescentado à trama original antes desta ser transmitida. O recetor ao receber a trama, utiliza a mesma função *f* e obtém, por sua vez, o bloco de bits (B1). Nessa altura, o recetor compara B com B1. Sendo iguais, a trama é considerada correta, caso contrário, significa tem erros e deve ser descartada.

Existem diversos métodos de deteção e correção de erros com menor ou maior complexidade. O método de deteção CRC (*Cyclic Redundancy Check*) usa o princípio enunciado acima, em que o bloco B1 deve ser zero, atendendo a que a adição do bloco B à trama original a tornou divisível por *f*. Este método, facilmente implementado em hardware, é usado em muitos protocolos de ligação lógica, nomeadamente em redes Ethernet[1] e Wi-Fi. Wi-Fi é a designação usada para a ligação em rede local sem fios, normalmente como sinónimo das normas IEEE 802.11a/b/g/n/ac.

### Protocolos de Acesso de Controlo de Ligação

Dois tipos de ligações comuns numa rede são as ligações ponto-a-ponto e as ligações multiponto, em particular, de difusão (*broadcast*). Uma ligação ponto-a-ponto envolve um nó emissor num extremo da ligação e um nó recetor no outro extremo. Ligações

[1] Atualmente, atendendo à baixa probabilidade de erro nestas redes locais, várias NIC Ethernet não geram o FCS por questões de desempenho.

de difusão envolvem vários nós que enviam e recebem através de um meio de difusão partilhado. Numa ligação de difusão, quando um nó envia uma trama todos os outros nós recebem essa trama. Exemplo de ligações de difusão são as redes locais baseadas em Ethernet partilhada ou redes sem fios (e.g., Wi-Fi).

Num meio partilhado, se não houver controlo ou coordenação no acesso ao meio pode haver colisões entre tramas transmitidas simultaneamente por dois ou mais nós. Quando há uma colisão de tramas é muito improvável que os recetores receberem corretamente as tramas transmitidas. Assim, um dos objetivos de um protocolo MAC (*Medium Access Protocol*) é coordenar o acesso ao meio de modo a reduzir ou eliminar a probabilidade de colisão de tramas, devendo os nós emissores envolvidos recuperar dessa situação.

Os protocolos MAC estão divididos em três categorias: protocolos de partição de canal, protocolos de passagem de ficha (*token-based*) e protocolos de acesso aleatório. Em particular, estes últimos são os mais usados nas redes locais comuns. As características e diferenças entre estes protocolos são estudadas nas aulas teóricas, não sendo diretamente objetivo deste trabalho.

Endereços MAC

A nível de ligação lógica, e em particular nas redes locais, os sistemas interligados são identificados por um endereço MAC. Um endereço MAC tem 48 bits de comprimento e é normalmente escrito em formato hexadecimal, por exemplo, 1A-23-F9-CD-06-9B. O endereço MAC é atribuído pelo fabricante da NIC (*Network Interface Card*) e não muda quando o nó muda de rede. Daí ser também designado como endereço físico. Pelo contrário, um endereço IP é um endereço lógico, i.e., depende da rede IP de acesso.

Normalmente, um nó terminal ou de interligação possui tantos endereços MAC quantas interfaces de rede ativas. Por exemplo, um *router* (apesar de operar sobre pacotes IP) tem também vários endereços MAC, um por cada interface de ligação disponível.

Quando um nó quer enviar uma trama na rede local insere os endereços MAC de origem e destino na trama. Numa rede local de difusão, Ethernet ou Wi-Fi, todos os nós da rede local recebem a trama. Cada nó recetor verifica se o endereço do destino MAC é igual ao seu. Em caso afirmativo, o campo de dados da trama (*payload*) é extraído e passado para o nível de rede, caso contrário, a trama é descartada. Há uma exceção: se o endereço destino for `FF-FF-FF-FF-FF-FF` (*broadcast*) todos os nós recebem e processam a trama.

*Address Resolution Protocol*

O principal objetivo do protocolo ARP (*Address Resolution Protocol*) é permitir fazer um mapeamento entre endereços do nível de rede (e.g. IP) e endereços nível de ligação lógica (MAC) por forma a possibilitar a entrega de dados entre nós adjacentes.

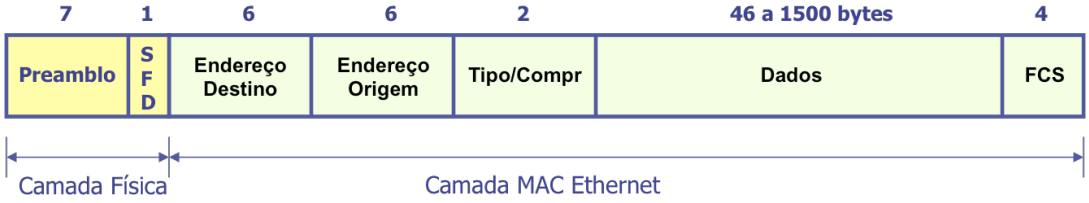
Suponha que um *host* na rede local quer enviar um datagrama IP para outro *host* na rede local. Suponha que conhece, provavelmente a partir do serviço de resolução de nomes – DNS, o endereço IP do *host* destino. Como sabe, o datagrama IP para ser enviado terá de ser entregue à camada de ligação lógica (L2) para ser encapsulado numa trama da tecnologia disponível e serializado para transmissão. A questão que se coloca é saber qual o endereço MAC destino a usar para enviar a trama que encapsula o datagrama IP, i.e., o *host* de origem vai ter de determinar o endereço MAC correspondente. Assim, sempre que necessário, o protocolo ARP permite obter o endereço MAC pretendido, através do uso das primitivas `arp-request` e `arp-reply`. Por cada resposta ARP recebida, e por questões de eficiência, cada nó da rede mantém uma tabela ARP (*cache*) que contém a correspondência entre endereços IP e os endereços MAC da rede local.

Note que o protocolo ARP tem um âmbito de operação restrito à rede local. Quando o destino IP é remoto, o protocolo ARP é usado para determinar o endereço MAC do *router* que está na mesma rede local, que, por sua vez, tem possibilidade de determinar qual o caminho que o datagrama IP deve seguir.

Ethernet

Ethernet é uma tecnologia de rede local bastante popular, havendo normas (*standards*) que permitem que a rede opere sobre diferentes meios de transmissão, topologias físicas e débitos de transmissão (tipicamente de 10Mbps a 10Gbps). A tecnologia Ethernet implementa um método de controlo de acesso ao meio que será detalhado nas aulas teóricas, e usa um formato de trama simples que inclui campos de controlo e um campo de dados.

Um trama Ethernet tem exatamente seis campos: (i) um campo para uma sequência de bits específica chamado *preâmbulo* (que o nó destino utiliza para sincronizar o seu relógio com o relógio do nó de origem e, assim, determinar quando começa a trama); (ii) o endereço MAC destino; (iii) o endereço MAC origem; (iv) um campo que indica o tipo de dados que a trama encapsula; (v) o campo de dados (*payload*); e (vi) o campo FCS (*Frame Check Sequence*) para o código de deteção de erros (CRC-32).




Interligação de Redes Locais

As redes locais são interligadas através de repetidores (*hubs)*, pontes (*bridges*) ou comutadores (*switches*) e *routers*.

Os *hubs* são dispositivos de interligação que operam a nível físico, i.e. repetem o sinal que chega através de uma porta de entrada para todas as outras portas.

Os *switches*, tal como as *bridges*, são dispositivos do nível de ligação lógica, processando tramas do nível de ligação. Um *switch,* com a ajuda de uma tabela de comutação, mantém para cada endereço MAC a indicação da interface de saída. Assim, quando uma trama Ethernet chega a uma interface é comutada de imediato para a interface apropriada. O preenchimento da tabela é feito através de um mecanismo de auto-aprendizagem. Quando chega uma trama a uma das suas interfaces, o *switch* examina o endereço de origem da trama e acrescenta uma entrada na tabela com o endereço MAC correspondente. Quando chega uma trama que o *switch* não consegue comutar com base na tabela de comutação difunde-a através de todas as suas interfaces de saída.

Por sua vez os *routers*, estudados no trabalho anterior, funcionam ao nível de rede encaminhando pacotes IP (datagramas) com base no endereço IP destino, i.e., de maneira parecida à forma como os *switches* lidam com os tramas. Para esse efeito, os *routers* utilizam uma tabela de encaminhamento que é atualizada manualmente com rotas estáticas ou automaticamente através da utilização de protocolos de encaminhamento tais como o OSPF (*Open Shortest Path First*).

As entradas da tabela de comutação de um *switch* têm um tempo de vida pré-definido após o qual são removidas se não chegarem tramas que refresquem essas entradas.

**3. Captura e análise de Tramas Ethernet**

A captura de tráfego deverá ser efetuada usando a aplicação Wireshark instalada na máquina nativa. Uma vez que as salas de aula atuais não disponibilizam uma ligação com fios a uma rede Ethernet, a captura será realizada na rede Eduroam. Este facto não impacta na realização do trabalho porque, por defeito, o Wireshark disponilibiliza o tráfego capturado ao utilizador como sendo (pseudo) Ethernet.

Assegure-se que a *cache* do seu *browser* está vazia.

Ative o Wireshark na sua máquina nativa.

No seu *browser*, aceda ao URL https://elearning.uminho.pt.

Pare a captura do Wireshark., e proceda da seguinte forma:

Localize o estabelecimento da conexão entre o cliente e o servidor HTTP (sequência de tramas com as TCP *flags* TCP SYN, SYN-ACK, ACK ativas).

Após a fase de estabelecimento seguro da conexão, obtenha o número de ordem da sequência de bytes capturada (coluna da esquerda na janela do Wireshark) correspondente à trama que transporta os primeiros dados aplicacionais enviados do cliente para o servidor (Application Data). Identifique também o número de ordem da trama com a resposta proveniente do servidor que contém os dados correspondentes ao acesso web realizado pelo cliente (*browser*).

Note que os dados aplicacionais são enviados de forma segura usando o protocolo TLS (*Transport Layer Security*), mapeados para um segmento TCP, transportado num datagrama IP que, por sua vez, é encapsulado no campo de dados da trama Ethernet. Expanda a informação do nível da ligação de dados e observe o conteúdo da trama Ethernet (cabeçalho e dados (*payload*)).

Responda às perguntas seguintes com base no conteúdo da trama Ethernet que contém a mensagem de acesso ao servidor (HTTP GET encriptada).

Sempre que aplicável, deve incluir a impressão dos dados relativa ao pacote capturado (ou parte dele) necessária para fundamentar a resposta à questão colocada. Para imprimir um pacote, use File->Print, escolha *Selected packet only* e *Packet summary line*, ou

use qualquer outro método que lhe pareça adequado para a captura desses dados. Selecione o mínimo detalhe necessário para responder à pergunta.

1. Anote os endereços MAC de origem e de destino da trama capturada.
2. Identifique a que sistemas se referem. Justifique.
3. Qual o valor hexadecimal do campo `Type` da trama Ethernet? O que significa?
4. Quantos bytes são usados no encapsulamento protocolar, i.e. desde o início da trama até ao início dos dados do nível aplicacional (`Application Data Protocol: http-over-tls`)? Calcule e indique, em percentagem, a sobrecarga (*overhead*) introduzida pela pilha protocolar.

A seguir responda às seguintes perguntas, baseado no conteúdo da trama Ethernet que contém o primeiro byte da resposta HTTP proveniente do servidor.

5. Qual é o endereço Ethernet da fonte? A que sistema de rede corresponde? Justifique.
6. Qual é o endereço MAC do destino? A que sistema corresponde?
7. Atendendo ao conceito de desencapsulamento protocolar, identifique os vários protocolos contidos na trama recebida.

**4. Protocolo ARP**

Nesta secção, pretende-se analisar a operação do protocolo ARP.

Verifique o conteúdo da *cache* ARP do seu computador.

- Windows: Digite `arp` ou `c:\windows\system32\arp` na linha de comando.
- Linux/Unix: O comando `arp` pode estar em vários locais, nomeadamente `/sbin/arp` (Linux), `/usr/sbin/arp` (FreeBSD) ou `/usr/etc/arp` (para outras variantes de Unix). O comando `arp` sem argumentos ou com a opção `-a` mostra o conteúdo da *cache* do seu computador (consultar `man arp`).

8. Observe o conteúdo da tabela ARP. Diga o que significa cada uma das colunas.

No sentido de observar o envio e receção de mensagens ARP, é conveniente apagar o conteúdo da *cache* ARP. Caso contrário, é provável que a associação entre endereços IP e MAC já exista em *cache*.

- Windows: os comandos `arp -d` ou `netsh interface ip delete arpcache` permitem apagar a *cache* ARP (use o *command prompt* como administrador).
- Linux/Unix: o comando `sudo arp -d -a` permite apagar a *cache* ARP.

Inicie a captura de tráfego com o Wireshark.

Para observar o protocolo ARP em operação, apague novamente a *cache* ARP (entretanto podem ter sido criadas novas entradas na tabela ARP) e assegure-se que o *cache* do browser está vazia.

Aceda a `https://alunos.uminho.pt` e efetue também um `ping` para um *host* da sala de aula que esteja a ser usado por outro grupo. Pare a captura de tráfego e tente localizar o tráfego ARP.

Se necessário, limite os protocolos visíveis apenas a protocolos abaixo do nível IP. Para tal, seleccione *Analyze->Enabled Protocols* e remova a seleção da opção IPv4 e IPv6.

Responda às seguintes perguntas:

9. Qual é o valor hexadecimal dos endereços origem e destino na trama Ethernet que contém a mensagem com o pedido ARP (`ARP Request`)? Como interpreta e justifica o endereço destino usado?
10. Qual o valor hexadecimal do campo tipo da trama Ethernet? O que indica?

11. Como pode confirmar que se trata efetivamente de um pedido ARP? Identifique que tipo de endereços estão contidos na mensagem ARP? Que conclui?

(Se necessário, consulte a RFC do protocolo ARP http://tools.ietf.org/html/rfc826.html.).

12. Explicite que tipo de pedido ou pergunta é feita pelo *host* de origem.

13. Localize a mensagem ARP que é a resposta ao pedido ARP efetuado.

    a. Qual o valor do campo ARP *opcode*? O que especifica?

    b. Em que campo da mensagem ARP está a resposta ao pedido ARP?

14. Na situação em que efetua um `ping` a outro host, assuma que este está diretamente ligado ao mesmo router, mas noutra subrede, e que todas as tabelas ARP se encontram inicialmente vazias. Esboce um diagrama em que indique claramente, e de forma cronológica, todas as mensagens ARP e ICMP trocadas, até à recepção da resposta ICMP do host destino.

**5. Domínios de colisão**

Uma rede local onde existam vários equipamentos ligados através de um meio partilhado comum constitui o que é denominado um domínio de colisão. Esta designação decorre da possibilidade de vários *hosts* poderem coincidir temporalmente no envio de uma trama, causando uma interferência mútua (colisão) que deteriora as tramas originalmente enviadas.

Num domínio de colisão, apenas um dispositivo pode transmitir num determinado instante e os restantes ficam à escuta para prevenir colisões. Por esse facto, a largura de banda é partilhada entre os diversos dispositivos. Na presença de uma colisão os dispositivos envolvidos têm que retransmitir a mesma trama Ethernet algum tempo depois. As normas Ethernet implementam um método de controlo de acesso ao meio denominado CSMA/CD (estudado nas aulas teóricas), que prevê a resolução de colisões.

Os domínios de colisão existem em segmentos de rede com equipamentos interligados via *hubs* partilhados (repetidores) e também em redes sem fios (Wi-Fi).

As redes atuais usam maioritariamente comutadores de rede (*switches*) para eliminar as colisões. Conectando cada dispositivo a uma porta do comutador, cada porta constitui um domínio de colisão (se a comunicação for *half-duplex*) ou são eliminados se a comunicação for *full-duplex*.

Ative o emulador CORE e carregue a topologia de rede com a solução de *subnetting* que construiu no âmbito do TP2. Substitua o *switch* do departamento A por um *hub (repetidor)*.

15. Através da opção `tcpdump` verifique e compare como flui o tráfego nas diversas interfaces do dispositivo de interligação no departamento A (LAN partilhada) e no departamento B (LAN comutada) quando se gera tráfego intra-departamento (por exemplo, fazendo `ping IPaddr` da `Bela` para `Monstro`, da `Jasmine` para o `Alladin`, etc.) Que conclui?

Comente os resultados obtidos quanto à utilização de *hubs* e *switches* no contexto de controlar ou dividir domínios de colisão. Documente as suas observações e conclusões com base no tráfego observado/capturado.

16. Construa manualmente a tabela de comutação do *switch* do Departamento B, atribuindo números de porta à sua escolha.

**Relatório do trabalho realizado**

O relatório do TP3 deve incluir:

- uma secção de "Questões e Respostas" do enunciado (inclua a questão, o *output* obtido (sempre que aplicável) e a resposta justificada).

- uma secção de "Conclusões" que autoavalie e resuma os resultados da aprendizagem nas várias vertentes estudadas no trabalho.

O relatório pode seguir o mesmo formato adotado no ensaio escrito (LNCS) ou um formato livre que facilite a inclusão dos resultados obtidos, e ser submetido na plataforma de e-learning **obrigatoriamente** com o nome RC-TP3-PL<TurnoGrupo>.pdf (por exemplo, RC-TP3-PL11.pdf para o grupo 1 do PL1, i.e., *group id* PL11) até ao ** final do dia da aula ** estipulada para conclusão do trabalho.