Criptografia (MIETI)

Módulo VI – Aplicações: Protocolos

Carlos Bacelar Almeida
jba@di.uminho.pt

Departamento de Informática
Universidade do Minho

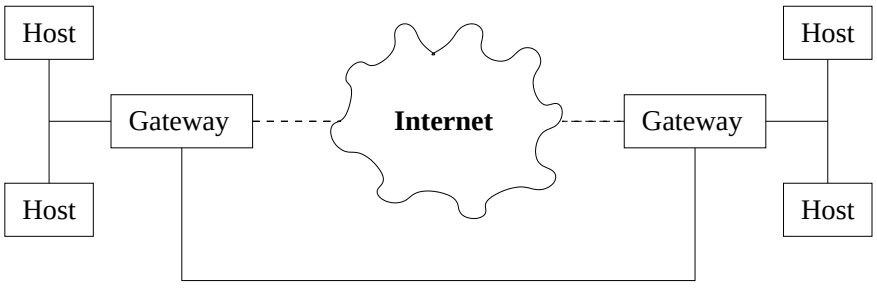
Introdução

- O IP (Internet Protocol) está ao nível da camada de rede do Modelo OSI. Fornece serviços de encaminhamento de pacotes através de redes heterogéneas.
- A maior parte das infraestruturas de comunicação na Internet são baseadas neste protocolo, conjuntamente com o TCP (TCP/IP).
- O IPsec fornece o mesmo conjunto de serviços, mas inclui funcionalidade extra ao nível da seguraça.
- Estes serviços são oferecidos ao nível da camada de rede, oferecendo protecção não só a esse nível, mas também a todas as camadas superiores.
- O IPsec está definido nas especificações RFC2401, e seguintes, da IETF.

- O IPsec oferece os seguintes serviços seguros ao nível da camada de rede:
  - Controlo de acessos
  - Integridade ao nível do pacote
  - Autenticação da origem de dados
  - Protecção contra pacotes repetidos
  - Confidencialidade
  - Confidencialidade de parte do tráfego
- Estes serviços permitem proteger ligações de rede entre nós IP, entre gateways seguras, ou entre um nó IP e uma gateway segura.
- Não substituem os serviços IP. São módulos adicionais que podem ser implementados e utilizados consoante o contexto e as necessidades das aplicações.

Revisão do Protocolo IP

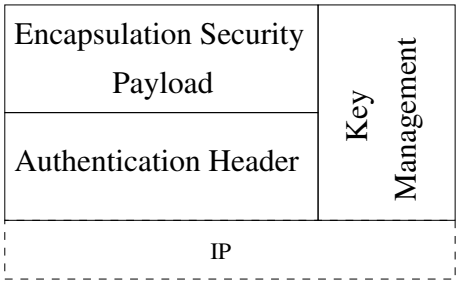
- Funcionamento:



  - Dentro de uma rede local, cada nó constroi um pacote IP incluindo os endereços de origem e destino no cabeçalho.
  - A comunicação com redes remotas é feita passando os pacotes a uma gateway: o endereço da gateway encapsula o verdadeiro.
  - A gateway substitui o encapsulamento reencaminhando o pacote. A gateway da rede remota retira o encapsulamento.

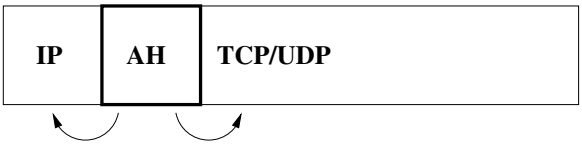
Estrutura do IPsec

- O IPsec está estruturado em duas sub-camadas:



- As duas sub-camadas são apoiadas por procedimentos e protocolos de gestão de chaves criptográficas (manuais ou automáticos).
- Os protocolos estão especificados de forma a serem independentes de algoritmos criptográficos. No entanto, alguns destes algoritmos estão pré-definidos.

IP Authentication Header



- A sub-camada **IP Authentication Header** (AH) inclui serviços de integridade ao nível dos pacotes, autenticação da origem de dados e, opcionalmente, protecção contra a repetição de pacotes.
- Recorre-se ao AH quando se pretende autenticação da informação correspondente à camada de rede (cabeçalho IP), ou quando a confidencialidade não é necessária (ou permitida).

Encapsulating Security Payload

- A sub-camada **Encapsulating Security Payload** (ESP) fornece confidencialidade (cifragem) da totalidade ou de apenas parte do tráfego.
- Como opções, o ESP oferece autenticação, verificação de integridade, e protecção contra pacotes repetidos. No entanto, esta funcionalidade abrange apenas informação correspondente às camadas de transporte e superiores (não trata o cabeçalho IP).
- No entanto pode ser, geralmente é, usado isoladamente.

Modos de Funcionamento

- Consoante o tipo de nós envolvidos, pode funcionar em:
  - **Transport Mode** – Sobre os cabeçalhos IP originais.
    - Apenas serve para ligações host-host
    - Com ESP não há protecção dos cabeçalhos IP (interferiria com a infraestrutura IP).
    - Com AH, há uma protecção parcial desses cabeçalhos.
  - **Tunnel Mode** – Sobre cabeçalhos IP encapsulados.
    - Corresponde a um túnel IP (caminho virtual entre nós).
    - Típicamente utilizado em ligações com/entre gateways
    - A protecção alcança todo o pacote original.
    - Diferenças entre AH e ESP mantêm-se, mas apenas para o cabeçalho exterior.
    - Para as camadas superiores, estas ligações aparecem como interfaces de rede adicionais.

Security Associations e SADs

- A gestão do IPsec baseia-se em **Security Associations**.
- Uma SA é uma ligação simplex identificada por três parâmetros incorporados nos cabeçalhos IPsec:
  - **IP Destination** Define o endereço de destino dos pacotes.
  - **IPsec Protocol** O protocolo (AH ou ESP) utilizado pela SA.
  - **Security Parameter Index** (SPI) Número de 32 bits que distingue SAs do mesmo tipo.
- Associados a este identificador estão todos os parâmetros de funcionamento necessários para a codificação e descodificação dos pacotes enviados através da ligação segura representada pela SA.
- Cada nó IPsec regista esta informação numa Security Association Database (SAD).
- Os pacotes que chegam a um nó são directamente procurados na SAD. A informação aí armazenada permite descodificar o pacote.

- Os pacotes que são transmitidos por um nó passam primeiro pela SPD. Caso o pacote deva ser transmitido utilizando IPsec,

Security Policy Database

- O IPsec funciona num nó (máquina ou gateway) de uma rede IP. Cada pacote que passa num nó IPsec pode ser tratado de acordo com uma de três políticas:
  - proteger o pacote com segurança IPsec,
  - enviar o pacote com IP simples, ou
  - descartar o pacote (no caso de gateways que limitam o tráfego entre redes).
- A política aplicada a cada pacote específico está armazenada numa Security Policy Database (SPD) que é gerida por um utilizador ou administrador do sistema.
- A pesquisa na tabela baseia-se em informação contida nos cabeçalhos de rede e transporte do pacote em questão: endereços IP de origem e destino, protocolo e portas de transporte (TCP/UDP).

Configuração do IPsec

- O IPsec permite grande flexibilidade na configuração:
  - dos serviços seguros que são utilizados e em que combinações;
  - da granularidade com que um determinado serviço é aplicado; e
  - dos algoritmos criptográficos utilizados em cada serviços.
- A configuração da granularidade de um serviço consiste em definir com que detalhe se distinguem os pacotes. Por exemplo:
  - podem-se cifrar todos os pacotes entre duas gateways, criando um canal seguro para todas as máquinas que utilizem essa ligação, ou . . .
  - podem cifrar-se apenas os pacotes que circulam entre duas máquinas protegendo apenas essa ligação.

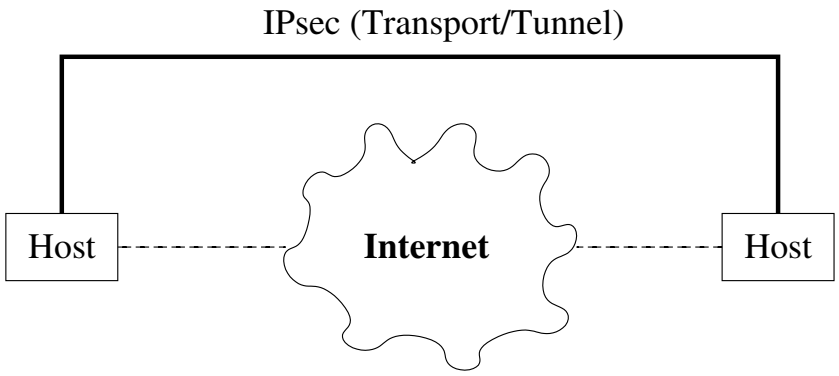
Gestão de Chaves e Algoritmos Criptográficos

- O IPsec permite o funcionamento baseado em técnicas manuais de gestão de chaves.
- No entanto, este tipo de operação não é adequado em sistemas com muitos nós, onde um sistema de criação dinâmica de SAs é preferível.
- Este tipo de funcionamento implica um sistema automático de gestão de algoritmos e chaves criptográficos.
- O sistema recomendado é o **Internet Key Exchange** (IKE), especificado nos RFCs 2407, 2408,2409 e 2412.
- O IKE permite a construção automática de SAs com negociação de parâmetros de comunicação e segurança, autenticação e protocolos de geração e acordo de chaves.

Tratamento de Pacotes Recebidos

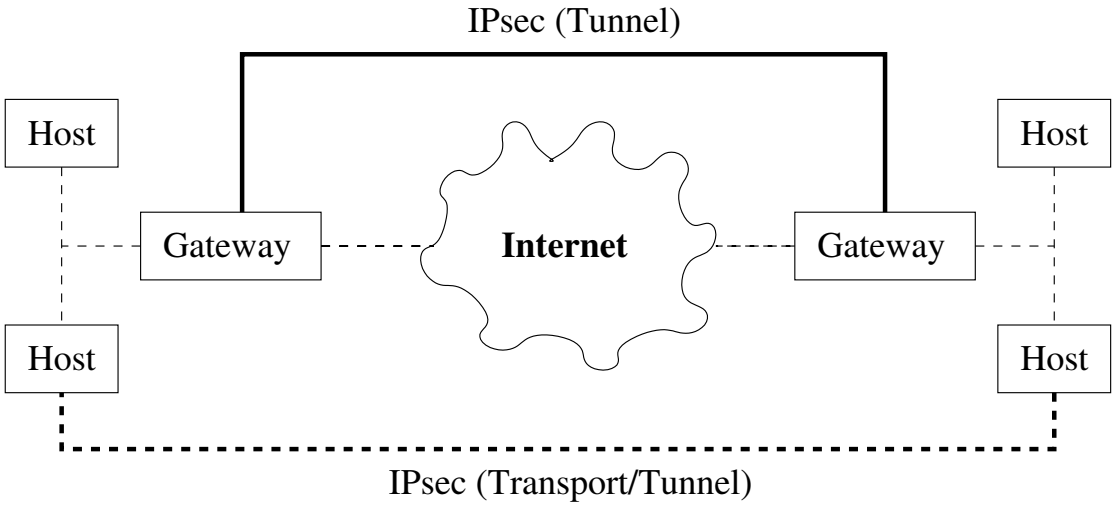
1. A informação contida no cabeçalho IPsec indica uma entrada na SAD:
2. Se essa entrada não existir desencadeia-se, se possível, a negociação de chaves para configurar uma nova SA. Se isto não for possível, o pacote é descartado.
3. Com base na informação associada à SA, processam-se os cabeçalhos e descodifica-se o pacote.
4. Consulta-se outra vez a SPD para saber se o pacote foi processado de acordo com a política correcta.
5. Se o processo decorrer sem problemas passa-se o pacote à camada de transporte.

Ligações Host-to-Host



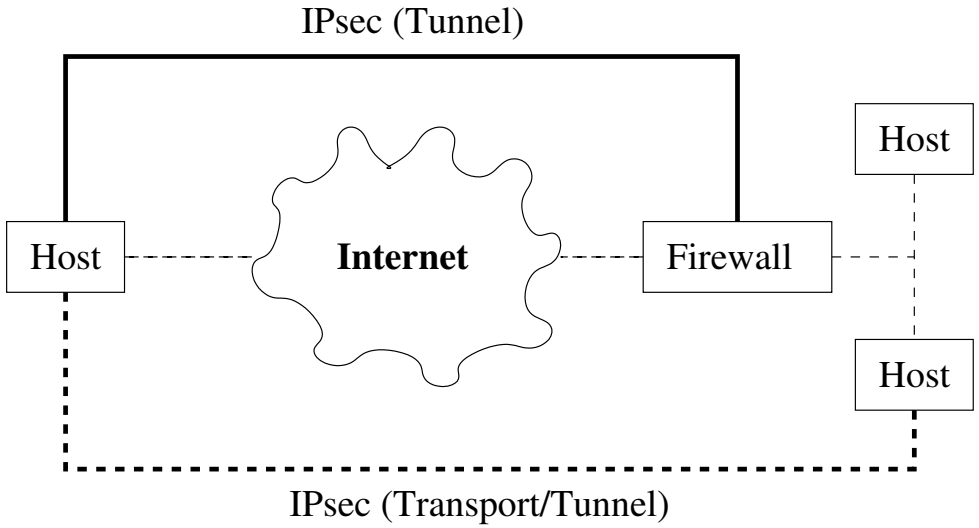
- Os pacotes trocados entre as duas máquinas são protegidos com IPsec, com base em Transport SAs.
- Pode ser utilizado o modo Tunnel, mas isso é redundante: o cabeçalho externo terá os mesmos endereços que o interno.

Virtual Private Networks (VPN)



- As gateways tornam a troca de pacotes entre redes totalmente transparente. Todo o tráfego é protegido.

Ligações remotas seguras (Road Warrior)



- Depois de a máquina externa se validar perante a firewall, passa a funcionar como se estivesse dentro da rede.

Críticas ao IPsec

- O IPsec é muito criticado pela sua complexidade (o seu desenvolvimento esteve a cargo de um comité).
- Os principais problemas apontados são:
  - A complexidade do IPsec dificulta a sua implementação e a sua aplicação. Além disso torna virtualmente impossível uma avaliação cabal da sua segurança.
  - A documentação é muito dispersa e difícil de ler.
  - Existe demasiada flexibilidade e funcionalidades aparentemente redundantes. Por exemplo:
    - Porquê dois modos de funcionamento (transporte e túnel): se o que se pretende é mais segurança, porque não utilizar apenas o túnel?
    - porquê dois protocolos (AH e ESP) quando seria muito mais simples especificar um protocolo único, mais simples e mais consistente?

- Um tão elevado grau de flexibilidade torna a configuração demasiado complexa para um utilizador comum, o que pode levar a instalações com graves falhas de segurança.
- Uma questão que levantou também alguma discussão foi o facto de a autenticação no IPsec ser feita depois da cifragem. Isto contraria o chamado “princípio de Horton”: *deve autenticar-se a informação necessária e suficiente para a interpretação de uma mensagem*. Um criptograma só pode ser interpretado depois de decifrado; não faz sentido autentica-lo. O argumento a favor disto é permitir o dismissal de pacotes inválidos sem perder tempo a decifra-los, reduzindo a hipótese de ataques por denial of service.
- O sistema de gestão de chaves recomendado (IKE) também contribui para as críticas ao IPsec: é demasiado genérico, as suas especificações estão mal escritas, etc.
- Conclusão: a segurança de um sistema IPsec depende demasiado das escolhas de implementação e configuração.

Ficha Técnica

- Algoritmos obrigatórios:
  - **Cifras Simétricas**: DES (CBC)
  - **Funções de Hash Criptográficas**: SHA-1, MD-5
  - **Message Authentication Codes**: HMAC
- Os protocolos são especificados de forma independente dos algoritmos, e há uma grande flexibilidade na utilização de outros algoritmos criptográficos.
- Está prevista por exemplo a utilização de diversas cifras simétricas (3-DES, IDEA, Blowfish), do RSA, do DSA, etc.

Introdução

- A Secure Sockets Layer está para o TCP como o IPsec está para o IP. É um *upgrade* da camada de transporte para incluir segurança nas comunicações.
- O SSL foi desenvolvido pela Netscape, e a sua versão 3 foi adoptada pela IETF sob a designação **Transport Layer Security** (TLS). O TLS está definido no RFC2246.
- Os serviços fornecidos pelo SSL incluem:
  - Confidencialidade baseada em cifras simétricas.
  - Autenticação baseada em criptografia de chave pública.
  - Integridade baseada em Message Authentication Codes.

Estrutura do SSL

- O SSL está estruturado em duas sub-camadas:

- A **Handshake Layer** permite a autenticação mutua entre clientes e servidores, e a negociação de algoritmos e chaves criptográficos antes de se iniciar a troca de dados através da Record Layer.
- A **Record Layer** encapsula a informação correspondente às camadas superiores.

Sessões SSL

- O funcionamento do SLL baseia-se em **sessões** estabelecidas entre um **cliente** e um **servidor**.
- Cada sessão SSL pode incluir várias ligações seguras, e cada nó pode manter diversas sessões SSL. Durante o seu estabelecimento e operação, as sessões e ligações SSL atravessam uma sequência de estados.
- Cliente e Servidor mantêm uma máquina de estados para cada sessão e ligação. A camada de Hanshake sincroniza os estados no cliente e no servidor.
- As transições entre estados efectuam-se em duas fases:
  - Primeiro constrói-se/negoceia-se um **pending state**.
  - Depois substitui-se o **operating state** pelo pending state.

Estado de uma Sessão SSL

- **Session identifier** Uma sequência arbitrária de bytes escolhida pelo servidor para identificar a sessão.
- **X509 certificate of the peer** Certificado do interlocutor.
- **Compression method** Algoritmo de compressão da informação antes de ser cifrada.
- **Cipher spec** Algoritmo de cifra simétrica (e algoritmo de hash criptográfico para utilização em MACs).
- **Master secret** Chave secreta partilhada por Cliente e Servidor e da qual são derivados todos os segredos utilizados na sessão (chaves e IVs).
- **Is resumable** Indica se a sessão pode ser utilizada para novas ligações.

Estado de uma Ligação SSL

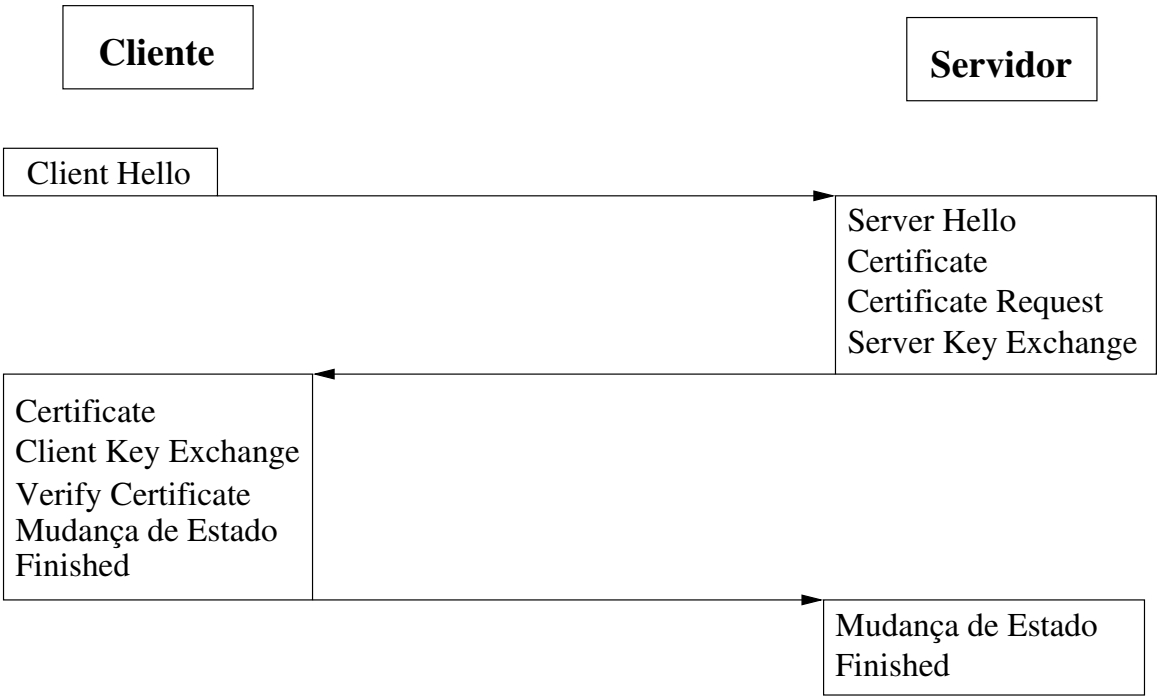
- **Server/Client random** Números aleatórios escolhidos por Cliente e Servidor para estabelecimento da ligação.
- **Server/Client write MAC secret** Chaves utilizadas por Cliente e Servidor para efectuar MACs sobre dados transmitidos.
- **Server/Client write key** Chaves utilizadas por Cliente e Servidor para cifrar dados transmitidos.
- **Initialization vectors** Vectores de inicialização (IV) para os modos de cifra simétrica que os utilizam.
- **Sequence numbers** Contadores sequenciais das mensagens enviadas e recebidas.

Record Layer

- Recebe informação arbitrária das camadas superiores, em blocos de dados de tamanho variável.
- Os dados são **fragmentados** em blocos com um máximo de 2^{14} bytes denominados **SSL Plaintext**.
- Os blocos SSL Plaintext são comprimidos com o algoritmo da sessão, originando blocos **SSL Compressed**.
- Os dados SSL Compressed são protegidos com a cifra e algoritmo de MAC definidos na CipherSpec da sessão (o MAC é calculado antes da cifragem). O resultado é um bloco do tipo **SSL Ciphertext**.
- Estes blocos são trocados entre Cliente e Servidor que têm de reverter estas transformações para obter o texto limpo.

Handshake Layer

- Os parâmetros de sessão e ligação utilizados pela Record Layer são estabelecidos pela Hanshake Layer.
- As mensagens da Handshake Layer viajam elas próprias sob o controlo da Record Layer. Inicialmente não há qualquer protecção: é utilizada uma *cipher spec* nula até que a primeira negociação seja concluida.
- Uma negociação é iniciada pelo Cliente com uma mensagem **Client Hello**. O Servidor deve responder com uma mensagem equivalente. Ficam acordados:
  - A versão do protocolo SSL a utilizar
  - O identificador da sessão e os números aleatórios.
  - Os algoritmos criptográficos a utilizar (os mais fortes dos suportados).
  - O algoritmo de compressão a utilizar



- Caso seja utilizada autenticação do Servidor, este envia o seu certificado X.509 ao Cliente, que o valida. Além da validação habitual, o Cliente assegura-se de que o nome de domínio do Servidor, indicado no certificado, está correcto.
- Parâmetros do Servidor específicos para acordo de chaves podem também ser enviados nesta fase (**Server Key Exchange**), se o seu certificado não incluir informação suficiente para esta funcionalidade.
- Caso o Servidor autentique o Cliente, solicita o certificado correspondente (**Certificate Request**). Este pedido inclui um desafio para ser utilizado na autenticação do cliente.
- O Servidor termina esta fase da negociação enviando uma mensagem **Server Hello Done**.

- Caso tenha recebido um pedido de certificado, o Cliente tem de envia-lo ou a negociação falha.
- Conjuntamente com o certificado o Cliente tem de enviar uma assinatura digital do desafio que recebeu, comprovando assim a posse da chave privada associada ao certificado.
- Finalmente, o Cliente envia os seus parâmetros para acordo de chaves (**Client Key Exchange**), altera o seu estado de sessão, e envia uma primeira mensagem cifrada que indica o seu estado de prontidão (**finished**).
- O Servidor efectua o mesmo procedimento e a negociação termina tendo sido acordado o Master Secret da sessão.

- A autenticação do servidor fica implicita pelo sucesso da comunicação cifrada nas mensagens **finished**, ou não?
- De facto, o servidor só fica autenticado se o protocolo de acordo de chaves implicar a utilização da sua chave privada.
- Isto acontece sempre:
  - No protocolo **RSAKeyExchange** o cliente gera um segredo e cifra-o com a chave pública do servidor. Para gerar o Master Secret, o servidor tem de decifrar este segredo com a sua chave privada.
  - Nos outros protocolos, os parâmetros públicos do servidor utilizados no protocolo de acordo de chave são assinados com a sua chave privada.

Segurança

- Mesmo se a versão actual do protocolo TLS é considerada segura, a verdade é que o historial de ataques a que a família de protocolos SSL/TLS se viu confrontada ao longo dos últimos anos é constrangedora! (ver `https://en.wikipedia.org/wiki/Transport_Layer_Security#Attacks_against_TLS.2FSSL`)
- Em geral, os ataques dirigidos ao protocolo propriamente dito foram sendo ultrapassados pelas revisões sucessivas (e.g. “ciphersuit rollback”, possível na versão SSL.2, foi ultrapassado com a autenticação das mensagens de *handshake* requerida pelo SSL.3)

- Os ataques mais recentes ao SSL/TLS dirigem-se maioritariamente a:
  - aspectos de implementação “menos cuidada” do protocolo (ou de *features* obscuras do protocolo, como no ataque **heartbleed**);
  - interacção com *cipher suites* específicas (e.g. utilização do RC4 ou *padding oracle attack* ao modo CBC).
- Certas implementações optam deliberadamente por suportar um fragmento restrito do protocolo para minimizar potenciais problemas de segurança (e.g. Amazon’s AWS S2N).
- Na prática, a enorme quantidade opções de configuração exige enorme cuidado no seu *deployment* (ver `https://ssllabs.com`)

Outros Protocolos

- *SSH (secure shell)* – protocolo de acesso remote a máquinas (inclui variantes seguras dos protocolos `telnet`, `ftp`, etc.);
- *PPTP*, *L2TP/IPsec*, *openVPN*, *SSTP* – protocolos para estabelecimento de *Virtual Private Networks*;
- ...