Administração de Sistemas ASIST

Tópico 1

O Administrador de Sistemas

Pinto Leite, Jorge (jpl@isep.ipp.pt)

O Administrador de Sistemas

- O que é e qual a necessidade de um Administrador de Sistemas?
- Qualquer instalação de sistema e infraestrutura necessita de cuidados quer no funcionamento normal, quer na intervenção se e quando ocorre um problema
- Para além disso, é razoável admitir que é conveniente a adoção de estratégias que permitam a recuperação dos dados em caso de necessidade
- É função do Administrador de Sistemas planear, implementar, verificar e agir em todas as vertentes de funcionamento e otimização da infraestrutura e dos sistemas
- Não esquecer que, dependendo da atividade da organização, pode também ser pertinente configurar os sistemas e infraestrutura para otimizar algum tipo de tráfego

O Administrador de Sistemas

- Numa organização pequena – no limite, o computador individual de cada um – a função de administração é mais simples
  - Embora por vezes alguns aspetos relacionados com a administração sejam descuidados…
- Mas à medida que a dimensão da infraestrutura e número de sistemas aumentam, mais responsabilidade e trabalho recai sobre o Administrador de Sistemas, tornando por vezes inviável que seja executado por uma única pessoa
- Apesar disso, há ou deve haver apenas um Administrador de Sistemas
- Porquê?

O Administrador de Sistemas

- Porque a responsabilidade não pode ser repartida por vários indivíduos!
- É função do Administrador de Sistemas
  - Planear, implementar, configurar e manter o bom funcionamento da infraestrutura
  - Planear, implementar, configurar e manter as aplicações inerentes ao funcionamento da organização
  - Planear, implementar, configurar e manter os requisitos de segurança (confidencialidade, integridade, disponibilidade)
  - Ser parte ativa nas políticas de segurança apropriadas para a organização
  - Analisar e selecionar novas tecnologias ou tendências que possam ser úteis
  - Estar atento e implementar a adaptação aos requisitos legais que devam ser cumpridos

O Administrador de Sistemas

- Em organizações muito grandes existem por vezes recursos humanos mais orientados para componentes da infraestrutura, sistemas e segurança
- Contudo, continua a haver necessidade de um Administrador de Sistemas único, que supervisiona e planeia todos esses recursos
  - Até porque a implementação ou alteração de um item pode condicionar quer a estratégia global da organização quer o desempenho de outros itens
- É pois o Administrador de Sistemas sempre o <u>único e exclusivo</u> responsável pela adaptação do funcionamento e desempenho da infraestrutura e sistemas aos requisitos da organização

O Administrador de Sistemas

- Tem ainda as responsabilidades acrescidas, de
  - Formar e treinar a equipa
  - Planear a estratégia de salvaguarda e recuperação de dados e aplicações
  - Assegurar o funcionamento pretendido, o que implica
    - Selecionar e adotar uma estratégia de reparação de equipamentos danificados
    - Resolver os problemas com as aplicações
  - Definir os critérios de gestão dos utilizadores
  - Assegurar o suporte técnico aos utilizadores
  - Planear e implementar a estratégia de atualização dos sistemas operativos e aplicações
- Várias destas tarefas podem ser automatizadas, mas compete-lhe analisar e definir a automatização, e, principalmente, validar que foram completadas com sucesso

O Administrador de Sistemas

- Compete-lhe ainda
  - Documentar todos os aspetos relativos aos sistemas e à infraestrutura
  - Elaborar os planos de recuperação do funcionamento
  - Elaborar os relatórios inerentes aos constrangimentos que possam eventualmente surgir

O Administrador de Sistemas

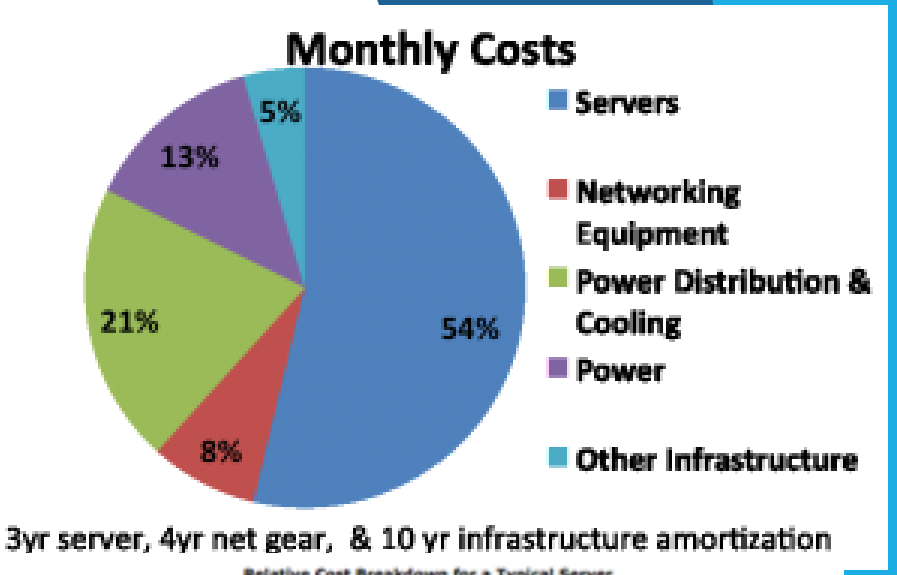
- Uma alternativa para a colocação dos sistemas que asseguram o funcionamento da infraestrutura tecnológica é de forma pontual, cada um no seu local
- Mas vimos já nas normas de cablagem estruturada que não deve ser esse o princípio a ser seguido
- O local de colocação dos sistemas e equipamentos a partir do qual se desenvolve a infraestrutura é designada habitualmente por **Centro de Processamento de Dados** (**CPD**)

O Administrador de Sistemas

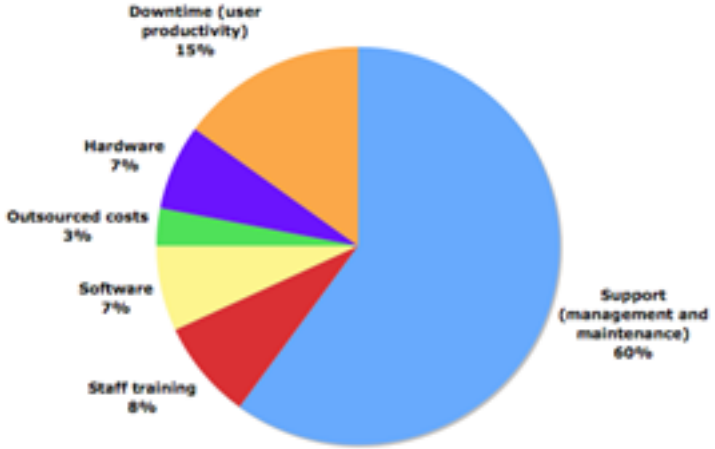
- As vantagens são evidentes
  - Tem-se assim um único local onde se deve assegurar os critérios ambientais (temperatura, humidade, poeiras, etc.) para umas melhores condições de funcionamento do ponto nevrálgico da infraestrutura
  - Torna-se mais simples a adoção e utilização de alternativas de alimentação elétrica em caso de falha (UPS, grupo eletrogéneo, etc.)
  - Possibilita a existência de piso falso e/ou teto falso para uma melhor passagem e distribuição das cablagens
  - Possibilita a existência e segurança de armários (bastidores, armários de telecomunicações) apropriados para conter os equipamentos, qualquer que seja o seu tipo
  - Único local onde se assegure a existência de segurança anti-incêndios

O Administrador de Sistemas

- Os custos operacionais do CPD crescem rapidamente em função dos equipamentos em atividade
- A área ocupada representa um custo para a organização
- Os custos com a adoção de equipamento diverso mas essencial aos requisitos pretendidos crescem à medida que mais novas tecnologias e marcas são adotados, acarretando maiores custos administrativos



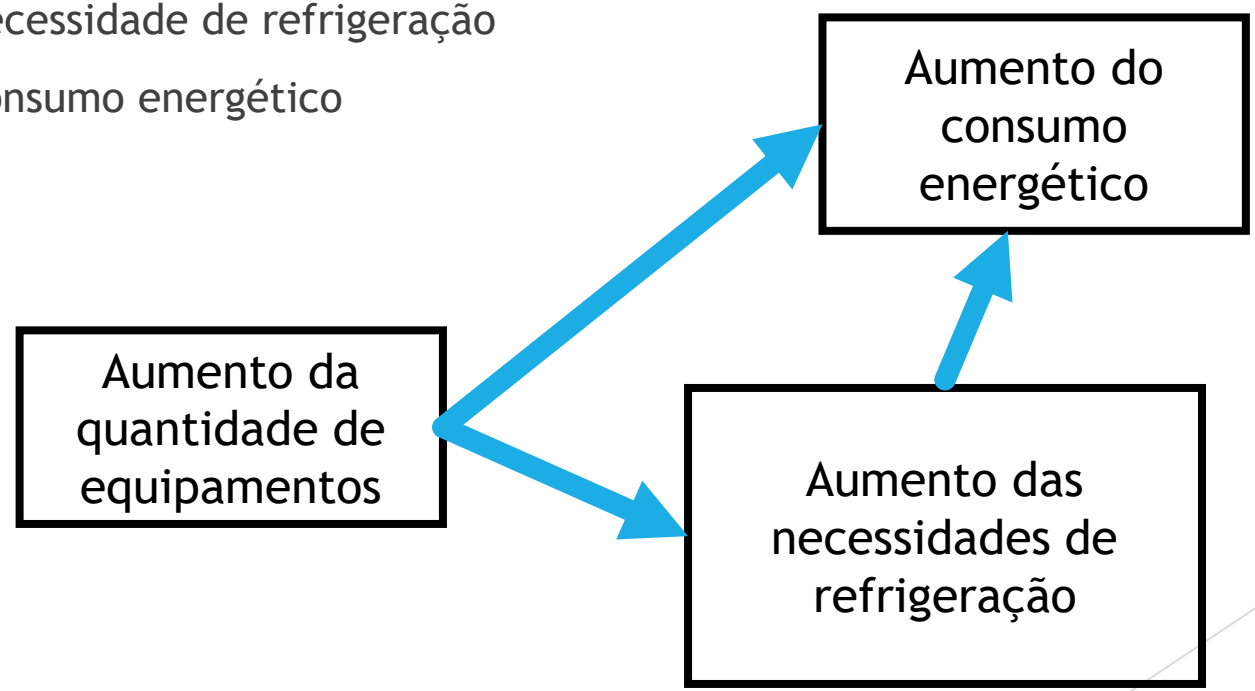
Relative Cost Breakdown for a Typical Server
(Source: IDC, 2007)



http://cloudscaling.com/blog/cloud-computing/understanding-cloud-datacenter-economies-of-scale/

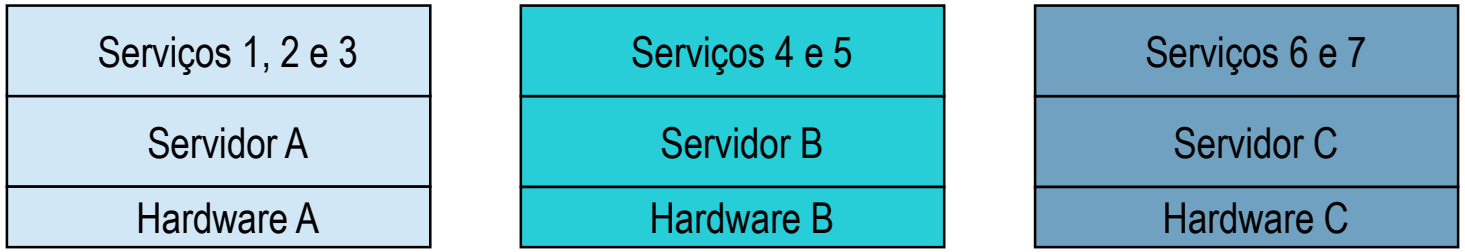
O Administrador de Sistemas

- Podem-se resumir os custos energéticos de um CPD em três fatores
  - Aumento da quantidade de equipamentos
  - Aumento da necessidade de refrigeração
  - Aumento do consumo energético



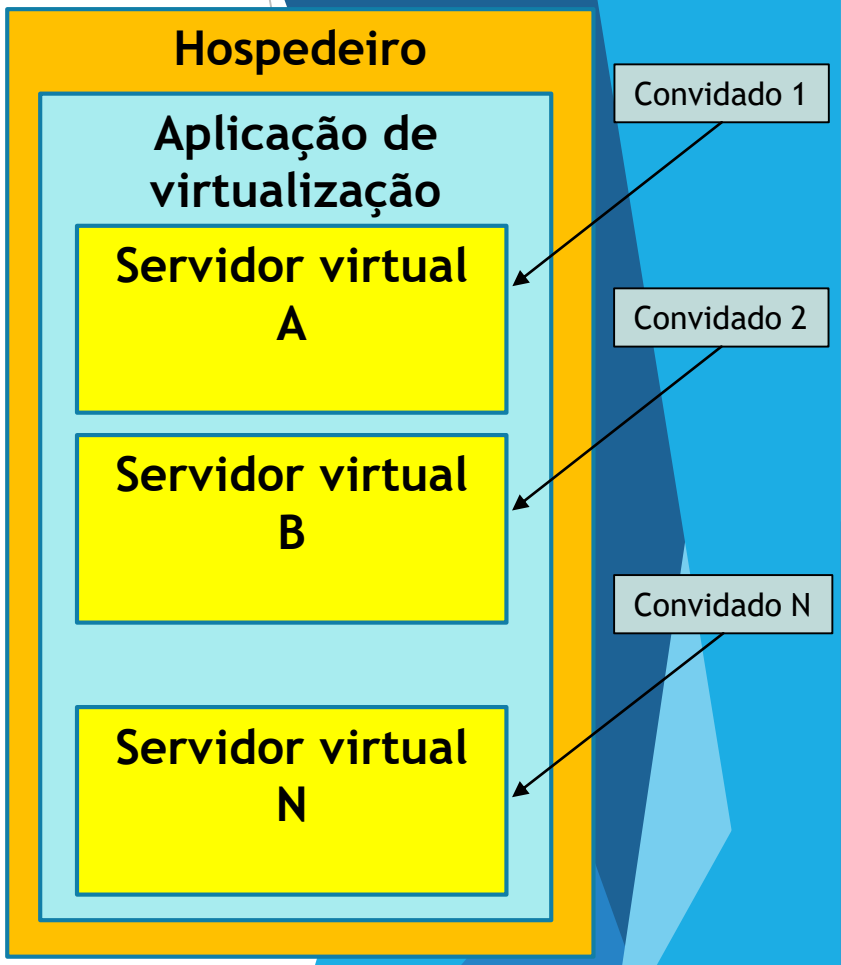
O Administrador de Sistemas

- Em consequência, há uma grande pressão para reduzir os custos operacionais, normalmente através de dois fatores
  - Os custos energéticos
  - A não adoção de uma política de *mais equipamentos para garantir mais serviços*
- A regra muitas vezes habitual era a instalação de novos equipamentos para assegurar um novo serviço (*não mexer muito no que funciona bem…*)
- Embora muitas vezes a taxa de utilização de cada servidor é reduzida ou até reduzidíssima



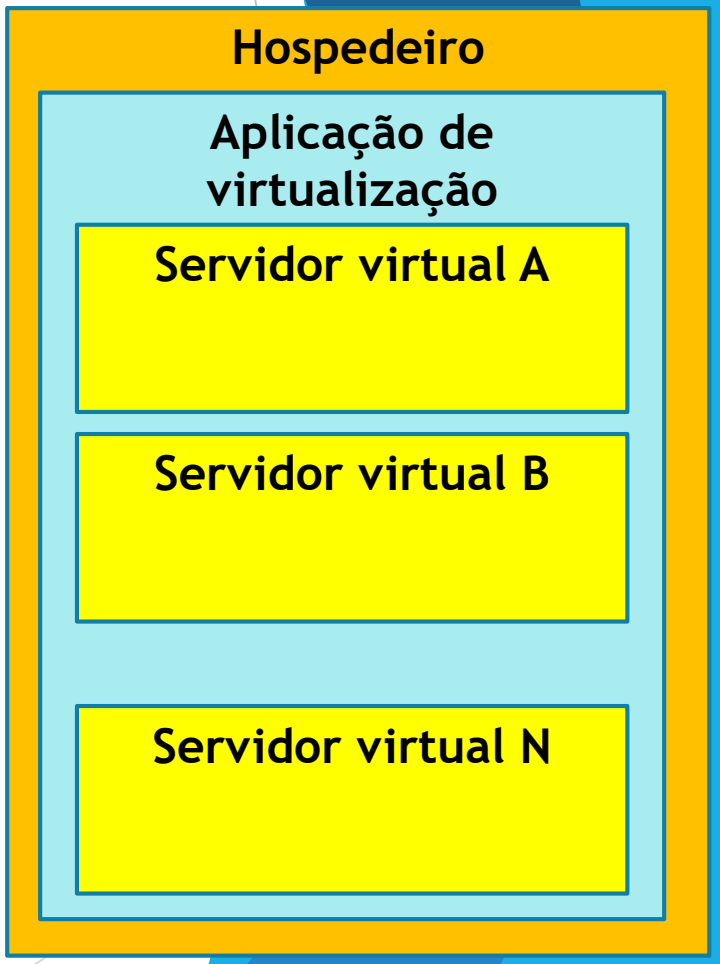
O Administrador de Sistemas

- Pode-se em vez desta estratégia recorrer à virtualização dos servidores
- Assim, existe no limite um só servidor (habitualmente – ou até, necessariamente - mais potente) que contém um hospedeiro de virtualização
- Neste hospedeiro são configurados e instalados servidores virtuais, designados por convidados
- A virtualização simula o equipamento físico com uma aplicação
  - Transforma o *hardware* em *software*



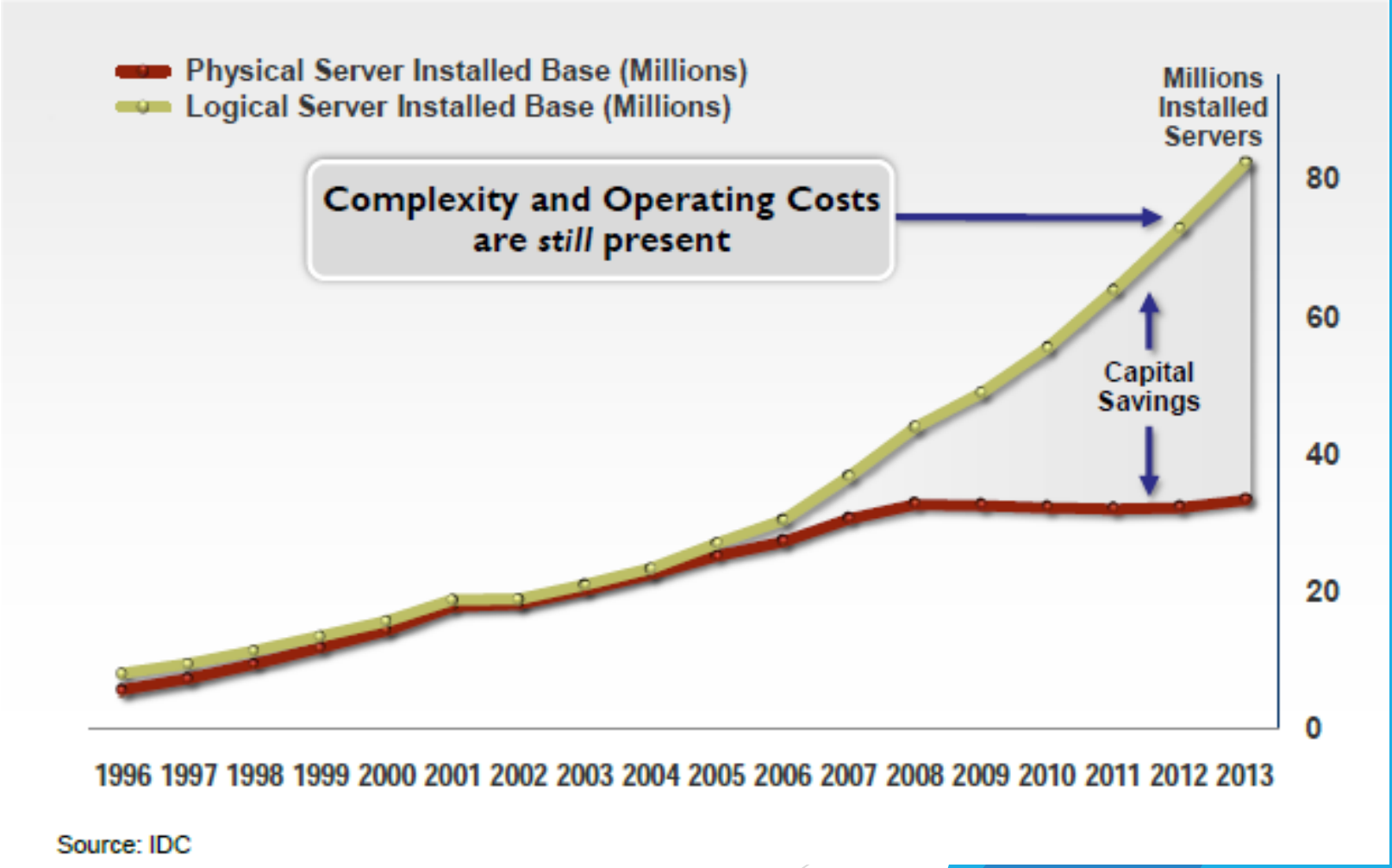
O Administrador de Sistemas

- De notar que cada convidado necessita do seu sistema operativo e a reserva de características (como disco, memória, placas de rede) que se socorrem das mesmas características físicas do hospedeiro
- Estas características são, contudo, distribuídas de forma dinâmica
- Os convidados funcionam de forma totalmente independente entre si
  - Embora possam ser criadas infraestruturas virtuais com ativos de rede que os interliguem
- Deve-se notar que a aplicação de virtualização limita o consumo máximo de recursos por parte de cada convidado, pelo que podem ocorrer limitações de desempenho



O Administrador de Sistemas

- Os custos administrativos ainda existem na virtualização, embora não cresçam na mesma proporção face aos custos associados à instalação de servidores físicos



O Administrador de Sistemas

- Mas sendo tudo uma aplicação que simula equipamentos físicos, a administração é realizada em exclusivo por manutenção e configuração dessa aplicação
- Operações como aumentar o disco, a memória, inserir mais processadores ou *cores* no processador, resume-se a uma configuração
  - Claro que algumas alterações implicam a posterior configuração no sistema operativo do convidado
- Facilita ainda alguns aspetos como as cópias de segurança, pois pode ser efetuada simplesmente através de uma imagem (*snapshot*)
- Principalmente, facilita imenso a produção em série de convidados com as mesmas características pois cada convidado pode ser copiado (*clonado*)

O Administrador de Sistemas

- A utilização da virtualização dos servidores
  - Aumenta a taxa de utilização do servidor físico hospedeiro (daí a necessidade de ser mais potente)
  - Reduz a quantidade de equipamentos físicos existentes
  - Simplifica as tarefas de administração
- Reduz desta forma os custos de administração e energéticos

O Administrador de Sistemas

- Em contrapartida, também tem inconvenientes
  - Se o hospedeiro ficar inoperacional, todos os convidados estarão também inoperacionais
  - Se o hospedeiro não for corretamente dimensionado, o desempenho dos convidados pode ser bastante afetado
  - Cada convidado deve ter a sua licença de sistema operativo que é totalmente independente da do anfitrião
  - É necessário prever para cada convidado o espaço em disco e memória necessário para o funcionamento do sistema operativo (e aplicações) que vão ser nele executadas
- A solução para o primeiro problema pode passar pela redundância de hospedeiros
  - Além de que a cópia ou localização de um convidado, caso haja mais do que um anfitrião, pode ser facilmente deslocada de um anfitrião para outro

O Administrador de Sistemas

- Entre os inconvenientes apontados aos servidores virtuais falámos da necessidade de ter o seu próprio sistema operativo
  - De notar que isto também pode ser encarado como uma vantagem, pois um hospedeiro, por exemplo, Windows pode alojar convidados Windows, Linux, etc.
- Contudo, por vezes é pretendido o mesmo sistema operativo em todos os convidados
  - Imagine-se por exemplo uma organização de desenvolvimento de aplicações
- Nesta situação é pouco razoável o recurso a máquinas virtuais sendo mais eficaz o recurso a *containers*

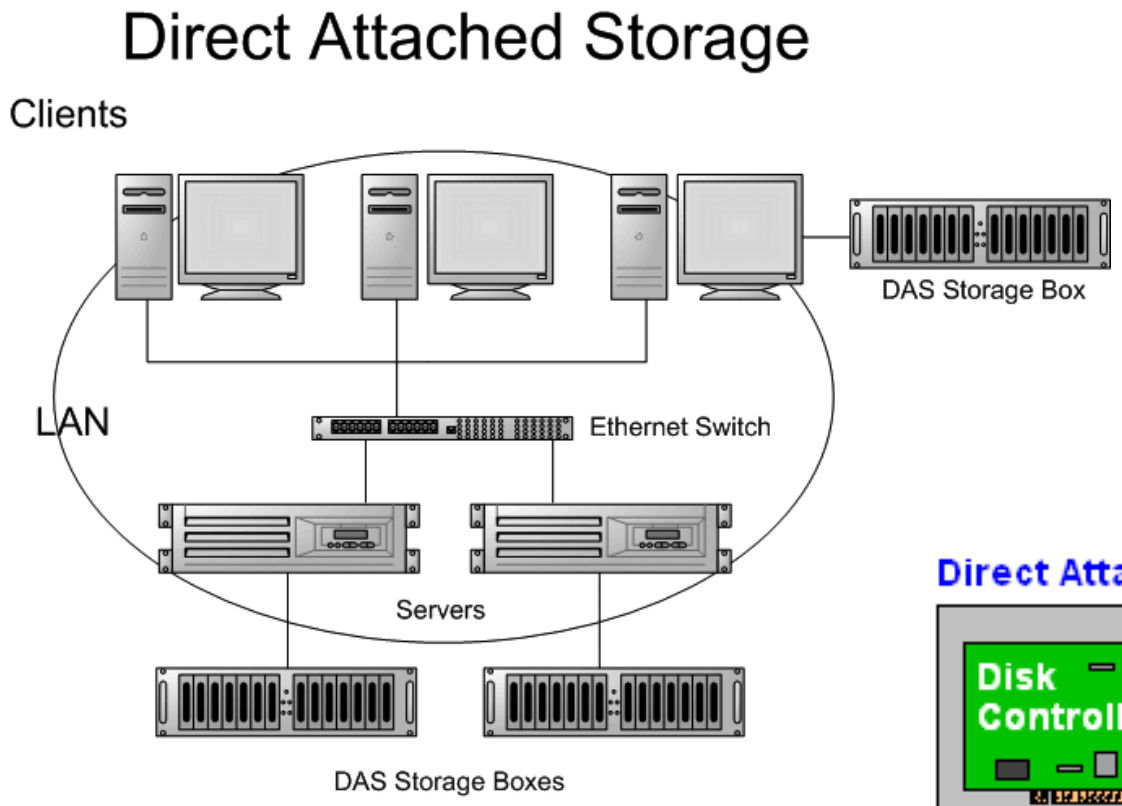
O Administrador de Sistemas

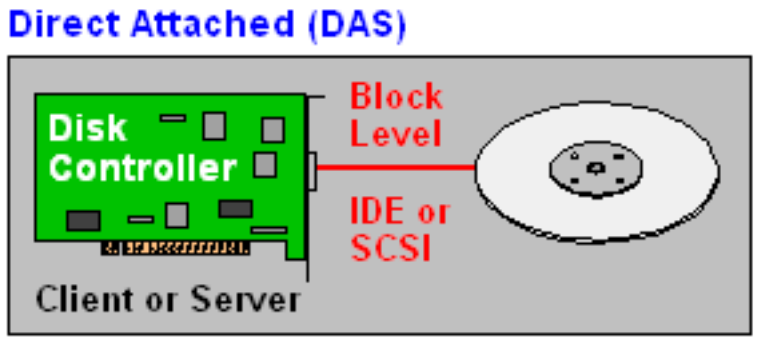
- Os *containers* são também uma virtualização de equipamentos físicos, contudo partilham com o hospedeiro parte do sistema operativo e bibliotecas
- Não é assim necessária a aquisição (e/ou instalação) de um sistema operativo em cada convidado
- Dada esta forma, o aproveitamento das capacidades do hospedeiro é mais eficaz, possibilitando mais convidados – por exigirem menos recursos físico
- Tem como inconveniente
  - Não permitirem sistemas operativos diferentes entre hospedeiro e convidados
  - Podem ocorrer problemas de segurança se um deles, hospedeiro ou convidado, for infetado com um *malware*
  - Mantém-se, claro, que a inoperacionalidade do hospedeiro acarreta a inoperacionalidade dos convidados

O Administrador de Sistemas

- Um outro aspeto a planear pelo Administrador de Sistemas é a localização dos dados
- Tipicamente, cada sistema possui o seu próprio armazenamento, designado por ***Direct Attached Storage*** (**DAS**) que conterá o sistema operativo, as aplicações e os dados
- O **DAS** apresenta-se ao sistema operativo como um *block device*
- Esta solução não é muito razoável pois é de utilização exclusiva do sistema a que está ligado
  - Claro que pode haver partilhas, mas o recurso a essa opção vai implicar outros custos
- A adição ou remoção de um dispositivo de armazenamento implica uma intervenção ao nível físico, muitas vezes com desativação temporário do sistema
- As operações de salvaguarda de dados são executados pelo próprio sistema, consumindo recursos
- Finalmente, não há economia de escala na partilha do armazenamento

O Administrador de Sistemas





Fonte: Computer Desktop Encyclopedia

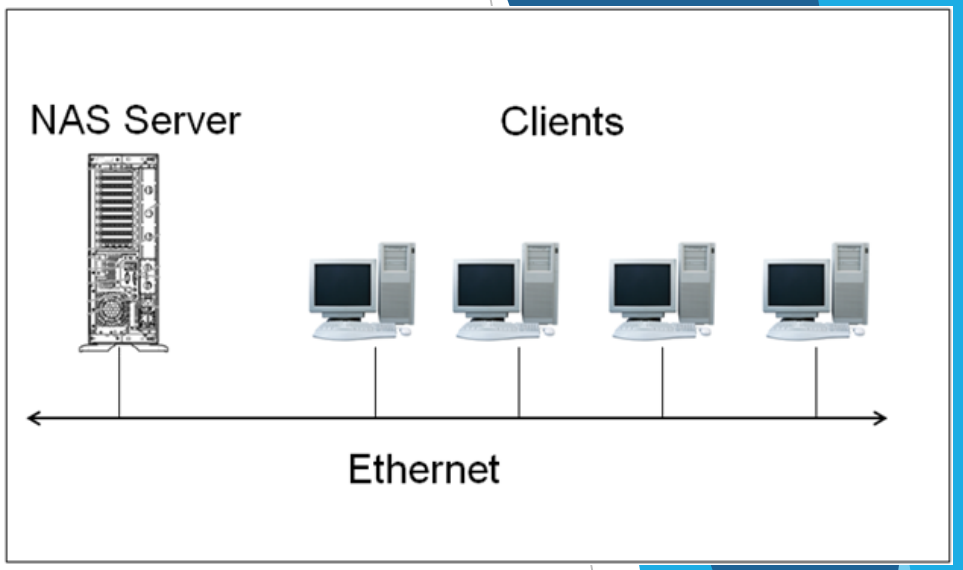
O Administrador de Sistemas

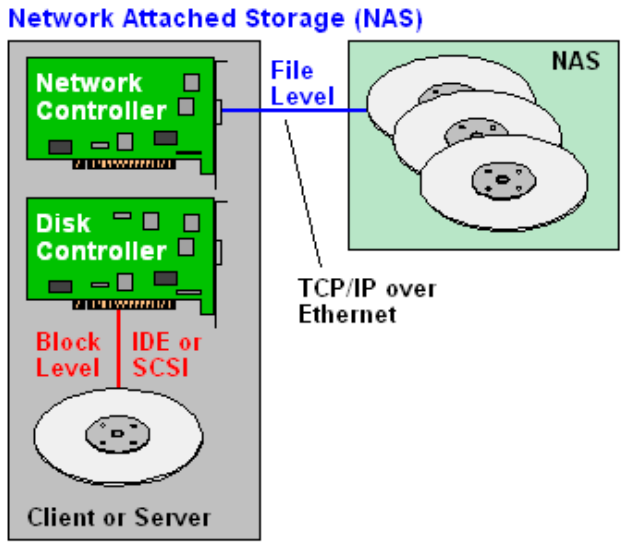
- Alternativamente pode-se utilizar uma ***Network Attached Storage*** (**NAS**)
- Este apresenta-se ao sistema operativo como um *file system*
- Apresenta várias vantagens
  - É uma forma económica de facultar um grande espaço de armazenamento a vários utilizadores e/ou sistemas
  - É rápido a instalar e configurar
  - Normalmente suporta níveis de **RAID** (***Redundant Access of Independent Disks***)
  - Permite definir permissões a pastas e ficheiros
  - Oferece uma elevada utilização dos recursos

  - *NAS is poor's man SAN*

O Administrador de Sistemas

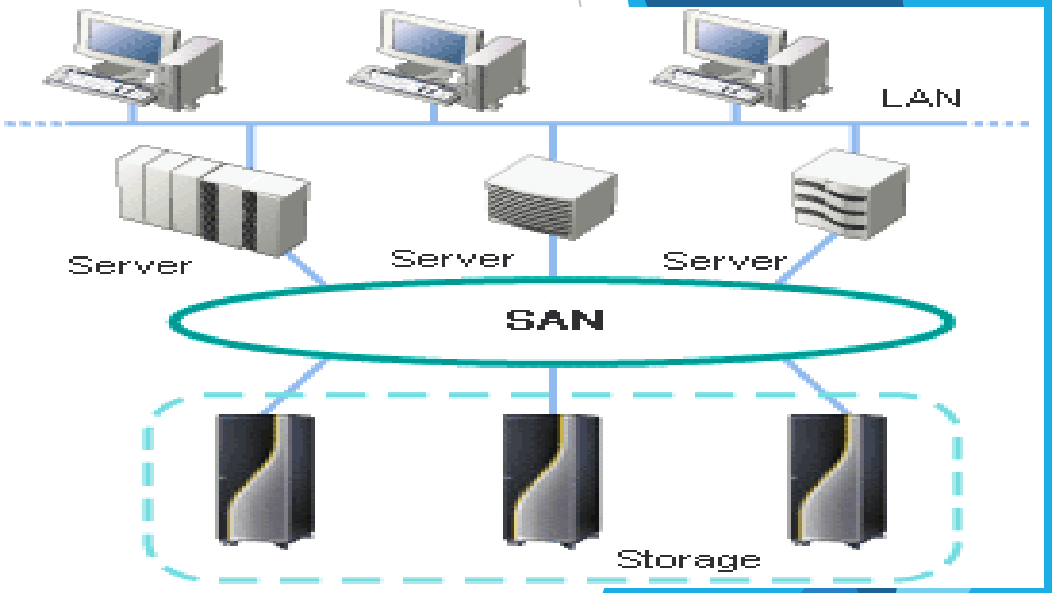
- Mas também tem inconvenientes
  - Ocupa recursos de rede (tem pelo menos um endereço IP)
  - Implica aumento da latência e potenciais problemas de transferência de dados
  - O desempenho é afetado pela disponibilidade da rede





O Administrador de Sistemas

- Uma outra alternativa, mais dispendiosa mas com maior capacidade é a ***Storage Area Network*** (**SAN**)
- Trata-se habitualmente de uma rede própria, fisicamente distinta para maior desempenho
- Deve disponibilizar uma elevada fiabilidade e segurança
- Pode ser montada com recurso a uma infraestrutura de rede Gigabit Ethernet, mas, idealmente, é suportada por tecnologia específica denominada ***Fibre Channel*** (**FC**)
- Também idealmente, deverá existir redundância de ligações na SAN

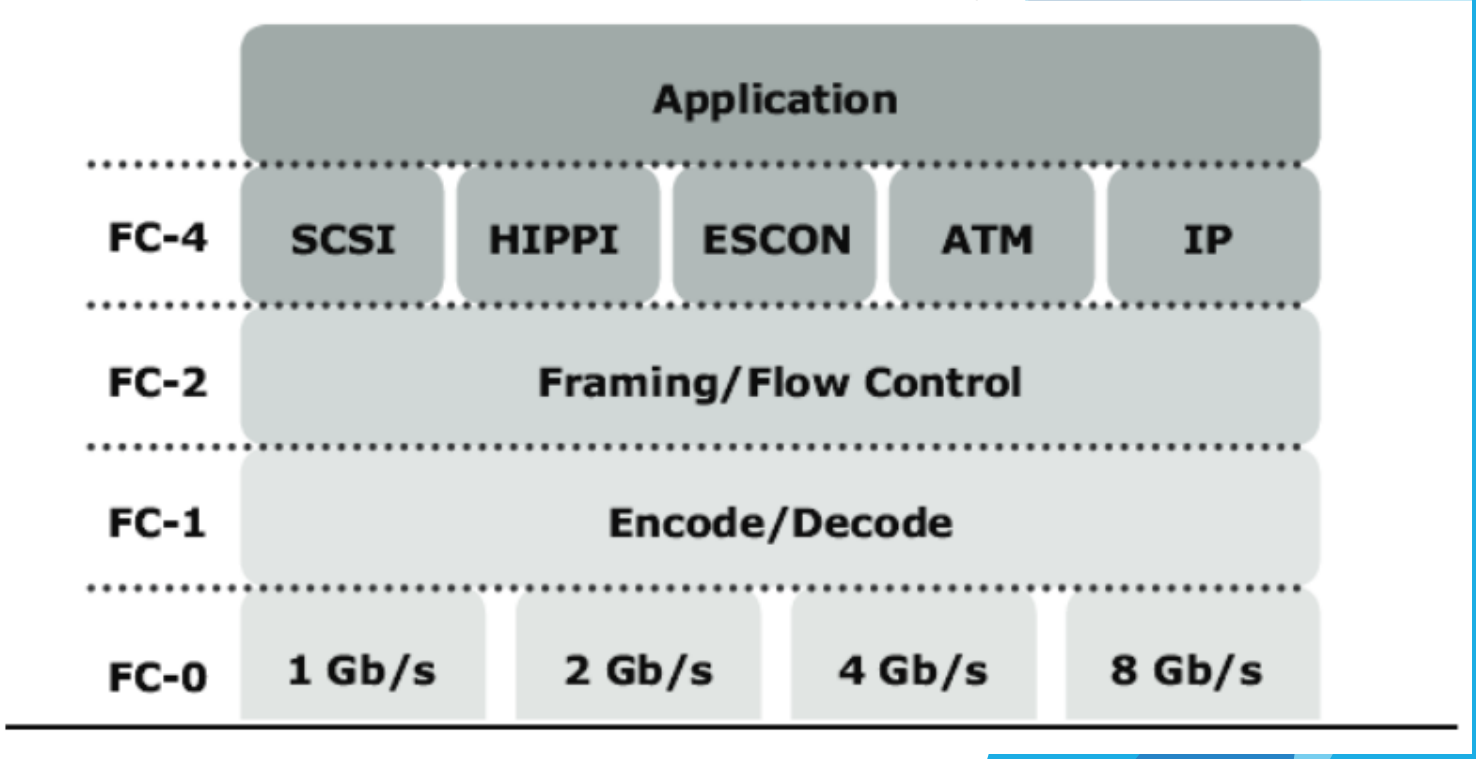


O Administrador de Sistemas

- Há dois protocolos e várias implementações de uma SAN
  - ***Fibre Channel Protocol*** **(FCP)**
  - ***Fibre Channel over IP*** **(FCIP)**
  - ***Internet Fibre Channel Protocol*** **(iFCP)**
  - ***Fibre Channel over Ethernet*** **(FCoE)**
  - *Non-Volatile Memory Express over Fibre Channel* (FC-NVMe)
    - Distingue-se por permitir a interconexão dos sistemas e discos SSD por uma ligação PCI Express de alta capacidade
  - ***Internet Small Computer System Interface*** **(iSCSI)**

O Administrador de Sistemas

- O protocolo FC tem 4 camadas apresentados na imagem
  - Na prática deveriam ser 5 mas um dos níveis (FC-3) nunca foi implementado



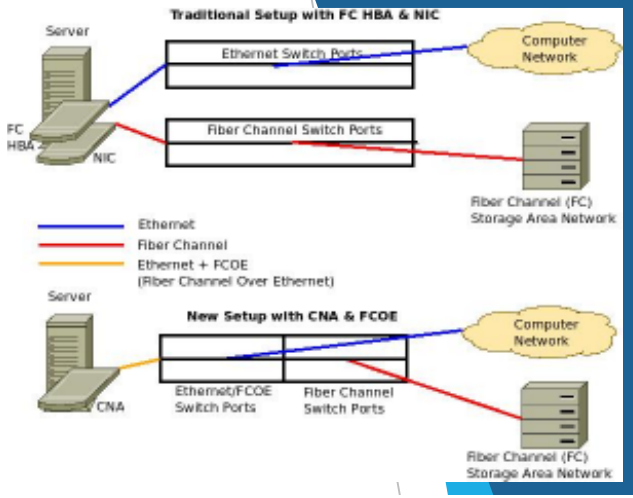
Fonte: JB Institute of Engineering & Technology

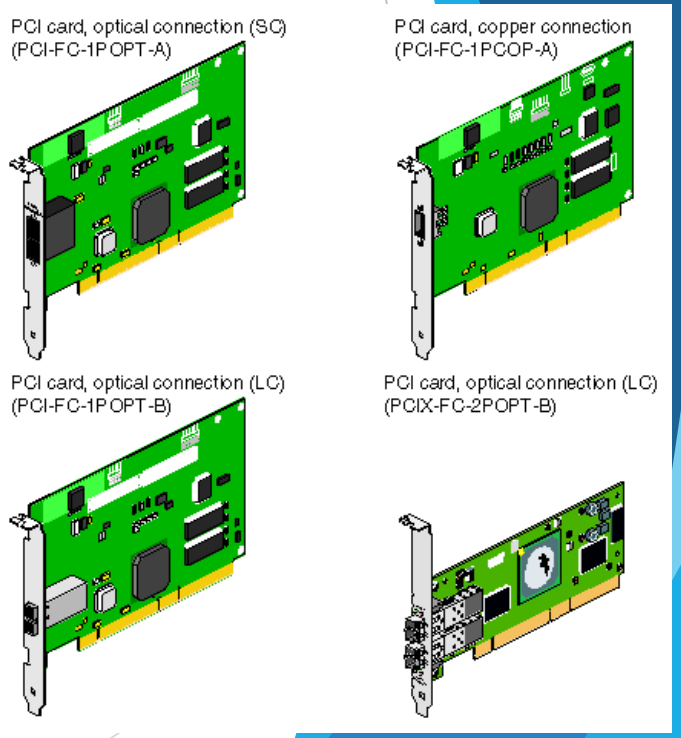
O Administrador de Sistemas

- FC-4
  - Define as interfaces de aplicação e o mapeamento para os níveis mais baixos
  - Alguns protocolos deste nível são o SCSI, ATM e IP
- FC-2
  - Nível de transporte
  - Contém os endereços de origem e destino, informação de controlo, etc.
- FC-1
  - Protocolo de transmissão
  - Inclui as regras de codificação e descodificação, controlo de erros
  - Na emissão são adicionados 2 bits a cada caractere que são retirados no recetor (para controlo de erro)
- FC-0
  - Nível físico

O Administrador de Sistemas

- A utilização de uma SAN com topologia Fibre Channel implica a existência de equipamentos específicos para interligação com a infraestrutura informática
- Salientam-se
  - ***Host Bus Adapter*** (**HBA**), placa que contém a interface apropriada para interligar o sistema a dispositivos de tecnologia não nativa
  - ***Converged Network Adapter*** (**CNA**) específica para interligar uma FCoE
    - Esta placa contém em simultâneo uma interface HBA e uma interface de rede
  - Switch / Gateway FC
- As topologias físicas são variadas, desde a ligação simples via um switch FC até à malha designada por **Core Edge**





O Administrador de Sistemas

- Há uma pressão enorme para maior capacidade, fiabilidade e desempenho do armazenamento de informação, motivado
  - Pela evolução tecnológica
  - Pelos requisitos de retorno de investimento
  - Pelo aumento exponencial de
    - Dispositivos e dados
    - Capacidade e necessidade de processamento de informação
    - Solicitações dos utilizadores
- As SAN necessitam atualmente de
  - Permitirem a interconexão
  - Serem mais escaláveis
  - Serem simples de gerir

O Administrador de Sistemas

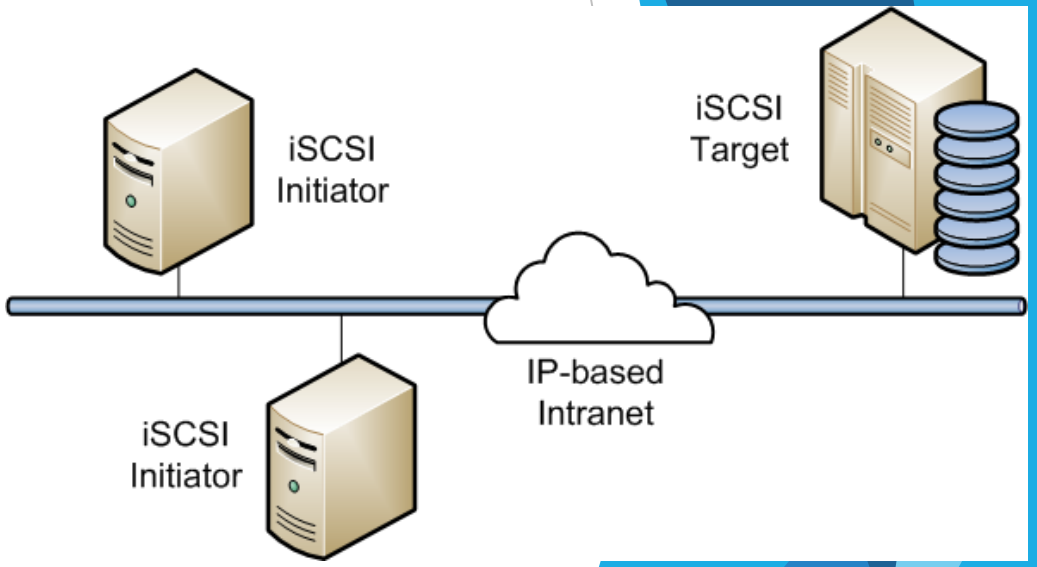
- Não admira por isso o aparecimento das tecnologias FCIP e iFCP
- Estas tecnologias permitem, em oposição ao FCoE, a interconexão de SAN remotas
- Note-se que o FCIP é implementado ao nível da própria SAN enquanto o iFCP é implementado ao nível do dispositivo
  - Como consequência não se pode interconectar diretamente uma SAN FCIP a uma SAN iFCP
- Ambas se baseiam e tiram partido do TCP/IP

O Administrador de Sistemas

- De notar que a tecnologia FC tem alguns constrangimentos, que se vão tentando resolver
  - Tipologia dos discos
  - Protocolos
- O primeiro foi resolvido com a progressiva substituição dos discos HDD por tecnologias mais recentes
- O segundo resolve-se com o NVMe
- A FC-NVMe consiste num protocolo e numa arquitetura que não se limita a discos SSD

O Administrador de Sistemas

- Uma alternativa mais económica é o iSCSI
- É nativo dos sistemas operativos modernos
  - Pelo que em oposição às outras tecnologias não necessita de placas ou dispositivos específicos
- Permite a interconexão de SAN remotas
- Baseia-se no TCP/IP
- Possui um *software initiator* (que tira partido do armazenamento) e um *software target* (que disponibiliza o armazenamento)
- O *initiator* acede ao *target* como se se tratasse de discos SCSI locais



Fonte: Oracle

O Administrador de Sistemas

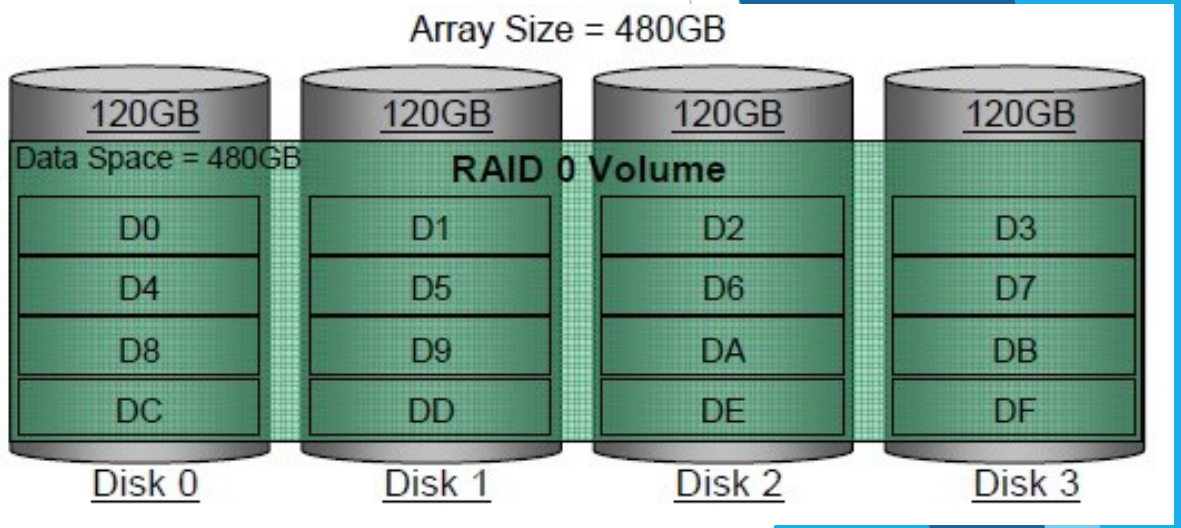
- Qualquer que seja o sistema de armazenamento adotado, o Administrador de Sistemas deve sempre prever os mecanismos necessários à mitigação ou eliminação dos dados
- Ora qualquer equipamento físico pode avariar, pelo que o armazenamento num disco único pode acarretar problemas em caso de avaria
- A estratégia de cópias regulares de segurança (que abordaremos no tópico 2) é essencial mas mesmo assim podem não ser suficientes para requisitos mais apertados
- Além disso, grandes volumes de dados podem ser demasiado extensos para estarem armazenados num único disco

O Administrador de Sistemas

- Surgiram por esta necessidade soluções que permitem
  - Replicar em tempo real um disco para outro
  - Expandir a capacidade de um disco utilizando espaço de outro(s) disco(s)
- Esta solução denomina-se ***Redundant Array of Independent Disks*** (**RAID**)
- Há vários níveis de RAID com finalidades distintas sendo identificados por número, mas vamos dividi-las aqui mediante um princípio base
  - Suportado diretamente pelos sistemas operativos mais recentes
    - RAID por *software*
  - Exigem controlador específico
    - RAID por *hardware*
  - De notar que a existência de um controlador específico não impede os tipos de RAID que não necessitam dele!

O Administrador de Sistemas

- RAID por software
  - **RAID 0** (RAID zero)
    - *Striping*
    - Dois ou mais discos trabalham de forma agrupada e simultânea, simulando para o sistema operativo um só espaço único de armazenamento que corresponde à soma do espaço individual dos discos
    - Tem como vantagem o aumento do espaço de armazenamento e melhoria de desempenho
    - Tem como inconveniente a perda da informação se um erro ocorrer em qualquer dos discos envolvidos

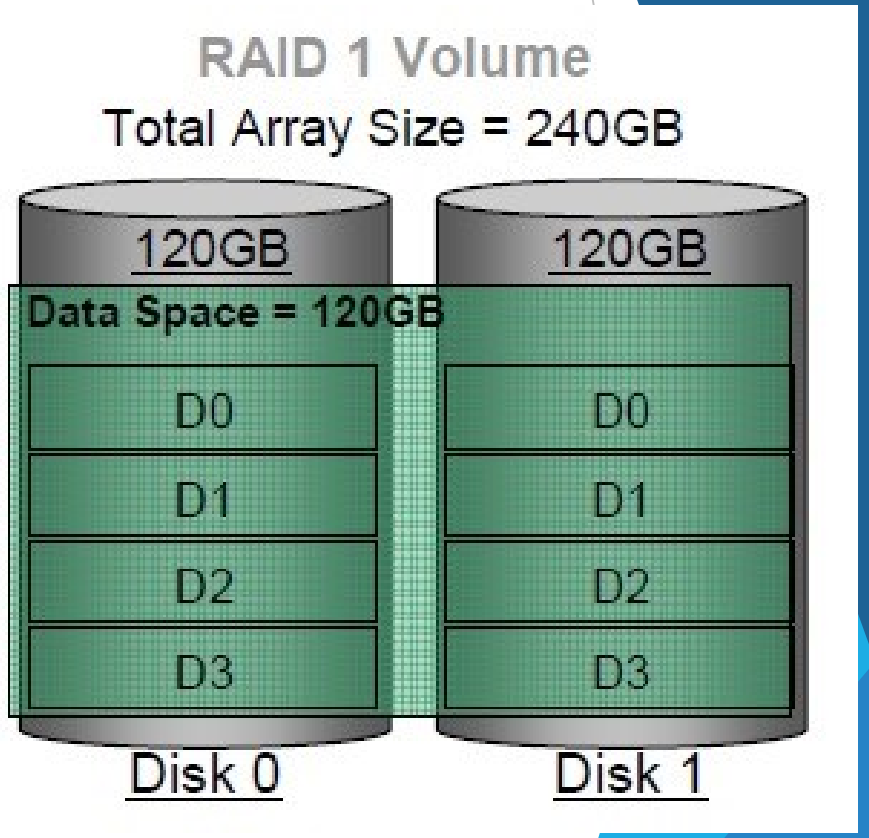


Fonte: intel.com

Espaço disponível = 120 x 4 = 480GB

O Administrador de Sistemas

- RAID por software
  - **RAID 1**
    - *Mirroring*
    - Duas unidades de disco são espelhadas, isto é, qualquer dado escrito numa delas é replicado na outra
    - Tem como vantagem
      - A capacidade de diminuir a probabilidade de perda de informação, já que os dados existem em dois discos distintos
      - O desempenho é melhor pois os dados podem ser lidos de qualquer dos discos – mas as gravações são mais lentas
    - Tem como inconveniente
      - Disponibilizar ao sistema operativo a capacidade de um disco só
      - Os discos devem ter a mesma dimensão

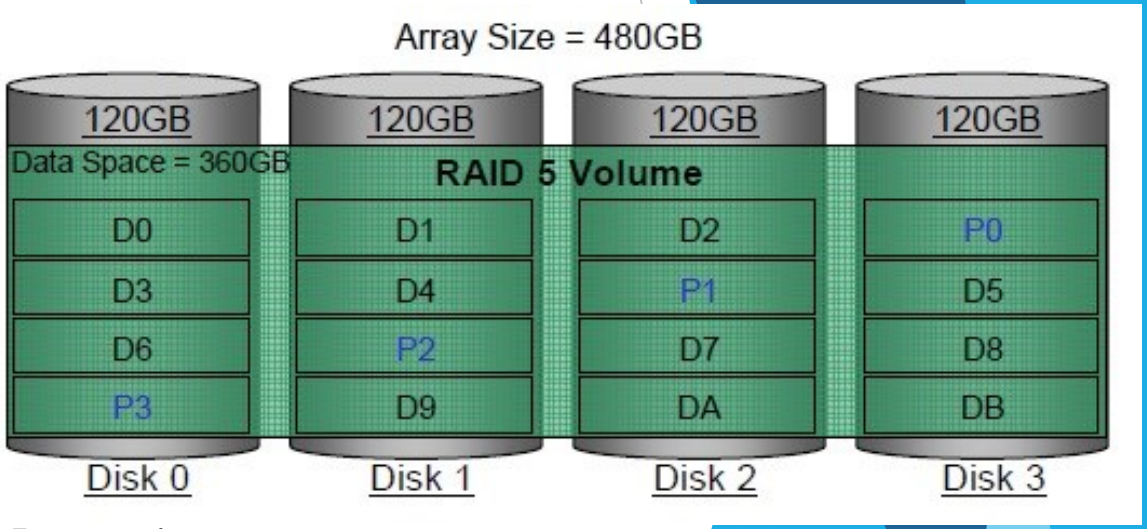


Fonte: intel.com

Espaço disponível = 120GB

O Administrador de Sistemas

- RAID por software
  - **RAID 5**
    - *Striping with parity*
    - Exige um mínimo de 3 (três) discos
    - Os dados são gravados em dois dos discos (Dx) e no terceiro é gravada uma informação de paridade (P*x*)
    - Tem como vantagem
      - O aproveitamento da capacidade (75% do disco é usado para dados)
      - O desempenho é melhor pois os dados podem ser lidos de qualquer dos discos – mas as gravações são mais lentas
      - Permite a recuperação da informação em caso de dano num qualquer disco
      - A capacidade máxima do conjunto aproveitável pelo sistema operativo é o tamanho do disco mais pequeno multiplicado pelo número de discos subtraído de um

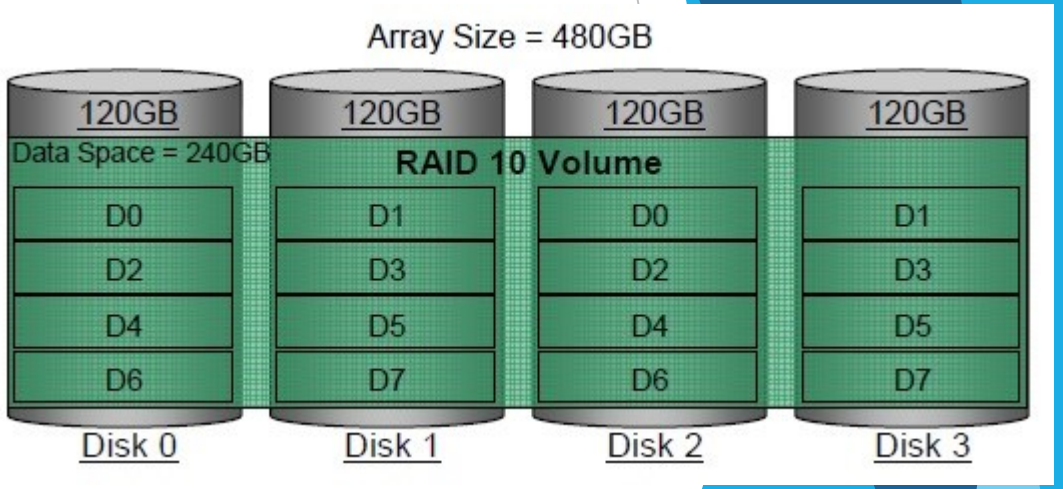


Fonte: intel.com

Espaço disponível = 120 x (4 – 1) = 360GB

O Administrador de Sistemas

- RAID por software
  - **RAID 10** (RAID 1 + 0)
    - *Striping with mirroring*
    - Dois dos discos são prolongamento (striping) um do outro, e o seu conjunto é espelhado (mirroring) no outro conjunto
    - Tem como vantagem
      - Combina as vantagens e desvantagens do RAID 0 e do RAID 1
      - A capacidade máxima do conjunto aproveitável pelo sistema operativo é o tamanho RAID 0 associado



Fonte: intel.com

Espaço disponível = 120 + 120 = 240GB

O Administrador de Sistemas

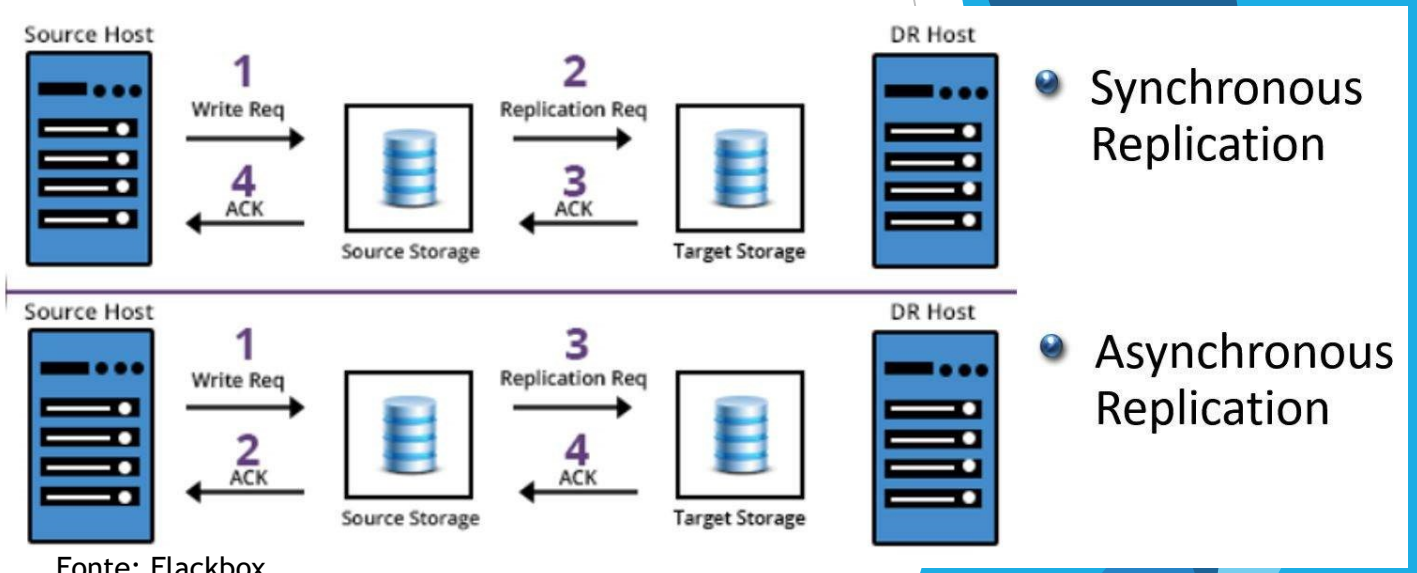
- RAID por hardware
  - **RAID 2**
    - *Striping* ao nível de bit
  - **RAID 3**
    - Striping ao nível de byte com paridade dedicada
  - **RAID 4**
    - Striping ao nível de bloco com paridade dedicada
  - **RAID 6**
    - Equivalente ao RAID 5
    - Tem dupla paridade distribuída
    - Permite recuperar a informação em caso de dano em dois dos discos
  - **RAID 50**
    - Exige um mínimo de 4 discos
  - **RAID 60**
    - Exige um mínimo de 8 discos

O Administrador de Sistemas

- A utilização de uma SAN é normalmente designada por **virtualização de *storage*** (porque não está fisicamente ligada a cada sistema)
- O sistema de armazenamento proporciona discos lógicos (***Logical Units*** **LUN**) aos servidores, na SAN, tirando partido dos protocolos apropriados
- Um disco lógico pode ser um disco físico, uma parte de um disco físico, ou até estar espalhado por diversos discos físicos
- O cliente acede ao disco lógico da mesma forma que a um disco local, e considera que tem acesso a toda a capacidade de armazenamento que lhe está atribuída

O Administrador de Sistemas

- Com o armazenamento virtualizado, a gestão e manutenção do armazenamento é efetuada por aplicações, não necessitando de paragem dos equipamentos
- Além disso, permite também a criação de instantâneos (*snapshot*), clonagem, e sincronização *(mirroring)* dos discos
- Estas operações veremos mais tarde que são cruciais para a recuperação em caso de desastre



Fonte: Flackbox

Administração de Sistemas ASIST

Tópico 2

Business Continuity Management

Pinto Leite, Jorge (jpl@isep.ipp.pt)

Business Continuity Management

- A utilização de recursos informáticos é uma realidade[1]
- Mesmo a nível particular, a utilização dos recursos informáticos é uma realidade cada vez mais premente[2]
- Esta realidade tornou essencial o (*bom*) funcionamento dos recursos informáticos
- Mas qual o impacto consoante a tipologia do utilizador?
  - É indiferente consoante se trata de um particular ou de uma empresa?

[1] Pordata, 17/8/2020: 99,2% das empresas com 10 ou mais trabalhadores (Portugal)
[2] Pordata, 17/8/2020: 3.641.758 assinantes de acesso à Internet (Portugal)

Business Continuity Management

- Não!
- O que muda é a expectativa e celeridade necessária para a sua utilização
- Os anseios e necessidades são as mesmas, o que altera é a capacidade (ou desejo) de manter o nível de funcionamento pretendido
- Tornou-se por isso premente a criação de uma metodologia para aferir o nível de funcionamento dos sistemas informáticos, e a definição de critérios que os meçam e/ou definam

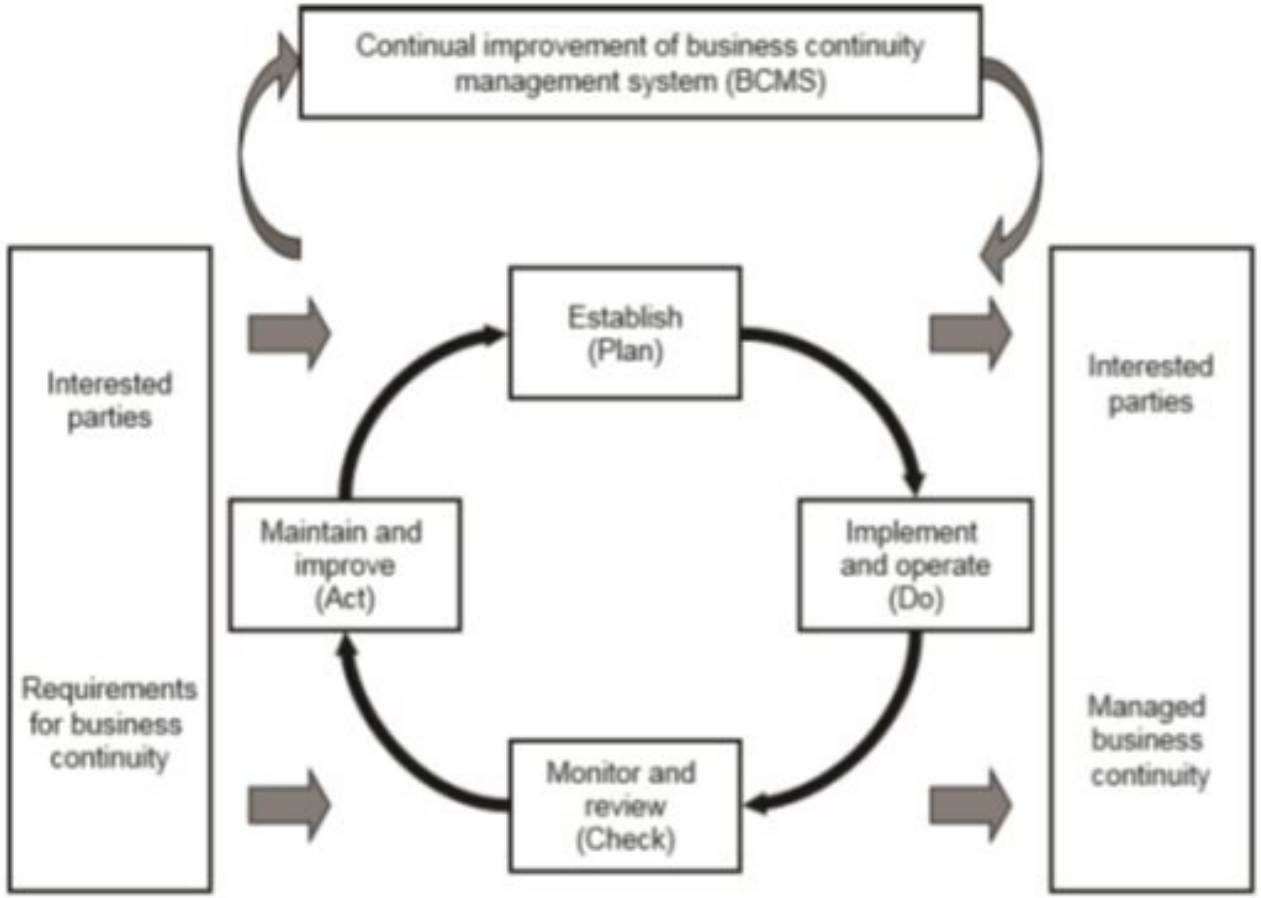
- Surgiram desta forma normas, das quais destacamos o ISO 27000

Business Continuity Management

- As normas ISO 27000, apoiadas por outras normas entre as quais a ISO 22301 e a 22313, definem critérios para manter a continuidade de negócio
  - Note-se que *negócio* deve ser entendido no sentido lato
- Elas definem regras tendo em mente organizações, mas adaptáveis à sua dimensão (que até pode ser um particular)
- O objetivo é normalizar metodologias e possibilitar desta forma critérios comuns
- Como traço comum, a necessidade de ser uma metodologia *top-down*, isto é, não faz sentido ser uma decisão individual de um só indivíduo sem o apoio dos decisores (numa organização, a gestão de topo)
- Estas normas definem o *Business Continuity Management* (BCM)

Business Continuity Management

- O modelo preconizado nas normas é o *Plan-Do-Check-Act* (PDCA)
  - Ou seja, é um modelo de contínua observação, planeamento e ação
- Como parte essencial para aplicação do modelo, integra um conjunto de critérios que devem ser cumpridos



Fonte: ISO 22301:2012

Business Continuity Management
Critérios I

- ***Maximum Tolerable Period of Disruption*** (MTPD)
- ***Maximum Tolerable Downtime*** (MTD)

- Estes dois critérios possuem um significado aproximado e definem:
  - MTPD: o tempo máximo de desempenho inferior aos requisitos da infraestrutura informática
  - MTD: o tempo máximo de inoperacionalidade da infraestrutura informática
- O objetivo é num período inferior ao definido por estes critérios a atividade e requisitos de desempenho da organização (isto é, *o negócio*) seja retomada
- Estes valores podem não ser fixos, estáticos
  - Por exemplo, pode haver um período temporal no qual é mais exigente do que noutro – por exemplo em negócios sazonais
  - Pode também haver um serviço mais premente do que outro
  - Nesse caso o BCM deve especificar todos os aplicáveis

Business Continuity Management
Critérios II

- ***Mininum Business Continuity Objective*** **(MBCO)**

- Especifica o nível mínimo de operacionalidade que deve ser mantida durante uma disrupção na infraestrutura
- Por exemplo, o serviço S1 pode não estar operacional mas o serviço S2 tem que se manter no nível de funcionamento aceitável
- O MTD/MTPD será certamente mais restrito para os serviços associados a este objetivo

Business Continuity Management

- Uma questão que se pode levantar aqui é o que é o nível de funcionamento aceitável
- Este é o nível de funcionamento (que veremos mais tarde) em termos de segurança, integridade e disponibilidade (ou seja, em termos de tempo de resposta) que é pretendido e aceite pela organização
- Designa-se este conjunto de parâmetros qualitativos e quantitativos por ***Service Level Agreement*** (**SLA**)

Business Continuity Management
Critérios I

- ***Businness Impact Analysis*** **(BIA)**
  - Identifica as atividades críticas da organização e as suas dependências; permite desta forma priorizar as operações de recuperação após uma disrupção

- ***Risk Assessment*** **(RA)**
  - Constituído pelos cenários que podem afetar a continuidade de negócio, a probabilidade de ocorrerem e o seu impacto

- ***Business Continuity Plan*** **(BCP)**
  - O BCP é parte integrante do BCM e documenta os procedimentos a efetuar para responder, recuperar, retomar e restaurar a um nível pré-definido de operação após a interrupção

Business Continuity Management
Critérios II

- O ***Risk Assessment*** (**RA**) é habitualmente representado por uma matriz de risco
- Cada item tem associada uma probabilidade e um impacto (severidade) estimado; o produto destes fatores dá uma medida para o risco

Risco = Impacto x Probabilidade

- Considerando probabilidades numa escala de 1 (menos possível) a 5 (mais possível) e impacto numa escala de 1 (marginal) a 4 (catastrófico), temos

Probability \ Severity	Catastrophic: 4	Critical: 3	Moderate: 2	Marginal: 1
Frequent: 5	High - 20	High - 15	High - 10	Medium - 5
Probable: 4	High - 16	High - 12	Serious - 8	Medium - 4
Occasional: 3	High - 12	Serious - 9	Medium - 6	Low - 3
Remote: 2	Serious - 8	Medium - 6	Medium - 4	Low - 2
Improbable: 1	Medium - 4	Low - 3	Low - 2	Low - 1

Fonte: Industry Safe

Business Continuity Management
Critérios III

- O *risk assessment* deve ser definido para todos os riscos identificados
- Pode não haver necessidade de construir a matriz de risco, mas tê-la auxilia a determinação dos riscos mais preocupantes e reduzi-los (mitigá-los)
- Mas importa que no BCM estejam definidos as ameaças riscos consideradas e a sua classificação de risco

Business Continuity Management
Critérios IV

- ***Disaster Recovery Plan*** **(DRP)**

- O DRP é também parte integrante do BCM
- Em situação normal, a infraestrutura funciona dentro dos parâmetros definidos no SLA e o BCM encarrega-se da observação (monitorização), planeamento, implementação das ações corretivas e demais atos considerados necessários
- Contudo, podem ocorrer problemas que impliquem a violação do SLA
- Mas será que violam também o MBCO ou o MTD?
  - Se o MBCO ou o MTD não são afetados, o BCM deve conter os procedimentos necessários para recuperar o SLA dos serviços em disrupção
  - Se são afetados, o DRP assume o controlo até à recuperação da situação pretendida

Business Continuity Management

- Sendo pretendida a continuidade de negócio, a elaboração do BCM deve analisar e mitigar constrangimentos potenciais
- Todas as ameaças e falhas que possam ocorrer devem ser avaliadas e, possivelmente, eliminadas – ou pelo menos mitigadas
- Comecemos por analisar os critérios SLA; deve contemplar não apenas os níveis de serviço (confidencialidade, integridade e disponibilidade) pretendidos mas também a forma de os medir e validar
- Um sistema que cumpre o SLA combinado designa-se por **sistema seguro**

Business Continuity Management

- Mas para que um sistema seja seguro, é necessário analisar e mitigar falhas que possam ocorrer
- Qual a probabilidade de falha (**P**) de um serviço?
  - O serviço depende de componentes, que por sua vez podem depender de outros componentes, etc. (veja-se como exemplo um serviço que corre num sistema, mas esse sistema depende da memória, do disco, da alimentação elétrica, etc.)
- A probabilidade de falha de um serviço dependente de vários fatores é dada pela soma das probabilidades de falhas de todos os fatores envolvidos

$$P = P_1 + P_2 + \ldots + P_N$$

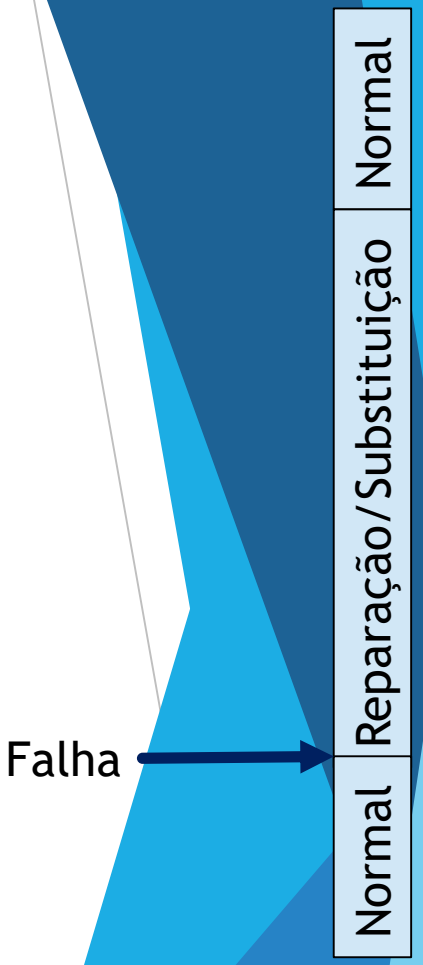
Business Continuity Management

- É contudo mais vulgar o uso do tempo médio entre falhas (***Mean Time Between Failures*** MTBF)
- Para o seu cálculo é tido em consideração não apenas o tempo médio para falhar (***Mean Time To Fail*** MTTF) mas também o tempo necessário para reparar ou substituir (***Mean Time To Repair/Replace*** MTTR)

MTBF = MTTF + MTTR

- Com base neste valor, podemos calcular a disponibilidade de um sistema ou serviço

$$\boldsymbol{Disponibilidade} = \frac{\boldsymbol{tempo\ de\ operação\ sem\ falha}}{\boldsymbol{tempo\ de\ operação\ total}} = \frac{\boldsymbol{MTTF}}{\boldsymbol{MTTF + MTTR}} = \frac{\boldsymbol{MTTF}}{\boldsymbol{MTBF}}$$



Business Continuity Management

- O inverso do MTBF é a **taxa de falhas por hora**, representada por λ ou, se calculado para um milhão de horas, a taxa de falhas por milhão de horas (**FIT**)
- O cálculo do MTBF de um sistema ou serviço que depende de vários subcomponentes em série pode ser calculado como o inverso da soma das taxas de falha por hora (ou por milhão de horas) de cada subcomponente
- Admita-se um sistema constituído por 3 (três) componentes não redundantes
  - Componente 1: MTBF1 = 20000 → FIT1 = 1000000/MTBF1 = 50
  - Componente 2: MTBF2 = 10000 → FIT2 = 1000000/MTBF2 = 100
  - Componente 3: MTBF3 = 15000 → FIT3 = 1000000/MTBF3 = 66,67
- O cálculo do FIT do sistema completo é **FIT1 + FIT2 + FIT3 = 216,67** pelo que

$$\mathbf{MTBF_{total} = 1000000 / FIT \approx 4615,3}$$

Business Continuity Management

- Para obter uma melhor disponibilidade é essencial aumentar o MTTF e diminuir o MTTR
- O aumento do MTTF pode ser obtido através de um controlo de qualidade dos produtos/serviços
- A diminuição do MTTR nem sempre é obtido com o mesmo controlo de qualidade dos produtos/serviços – para além de ser dependente do tipo de produto/serviço em causa
- A reparação de um serviço que interrompe o seu funcionamento não é comparável com a substituição de um sistema!
- Designa-se por **alta disponibilidade** a obtenção de uma disponibilidade de 100% ou próximo desse valor

Business Continuity Management

- Entre os princípios que podem ser ativados para aumentar a disponibilidade ressalta a **redundância**
- Contudo, um aspeto a ter em conta é a total independência e autonomia dos componentes redundantes
- O cálculo do MTBF de um sistema redundante socorre-se da taxa de falhas por hora, λ, dos seus componentes
- Admita-se um sistema constituído por 3 (três) componentes redundantes mas não redundantes entre si
  - Componente 1: MTBF1 = 20000 → $\lambda 1$ = 1/MTBF1 = 0,00005
  - Componente 2: MTBF2 = 10000 → λ 2 = 1/MTBF2 = 0,001
  - Componente 3: MTBF3 = 15000 → λ 3 = 1/MTBF3 ≈ 0,000067
- O cálculo de λ do sistema completo é **$\lambda 1 \times \lambda 2 \times \lambda 3 \approx 0{,}0000000000003$** pelo que

$$\mathbf{MTBF_{total} = 1 / \lambda \approx 3 \times 10^{12}}$$

Business Continuity Management

- Através da **prevenção de falhas** (*Fault Avoidance*) procura-se evitar a ocorrência de falhas que coloquem em causa a continuidade de negócio
- Vários mecanismos podem ser utilizados, dos quais se destacam a monitorização e alarmística (verificação da utilização de recursos como disco, CPU, controlo da temperatura, humidade, etc.)

- O nível de **tolerância a falhas** (*Fault Tolerance*) classifica o sistema ou serviço de acordo com o seu comportamento perante uma falha de qualquer origem
  - Se mantém o funcionamento: ***Full Fault Tolerant***
  - Se ocorre uma degradação temporária: ***Gracefull Degradation***
  - Se a degradação é significativa ou prolongada: ***Fail Soft***
  - Se torna o sistema/serviço indisponível mas mantém a integridade: ***Fail Safe***

Business Continuity Management

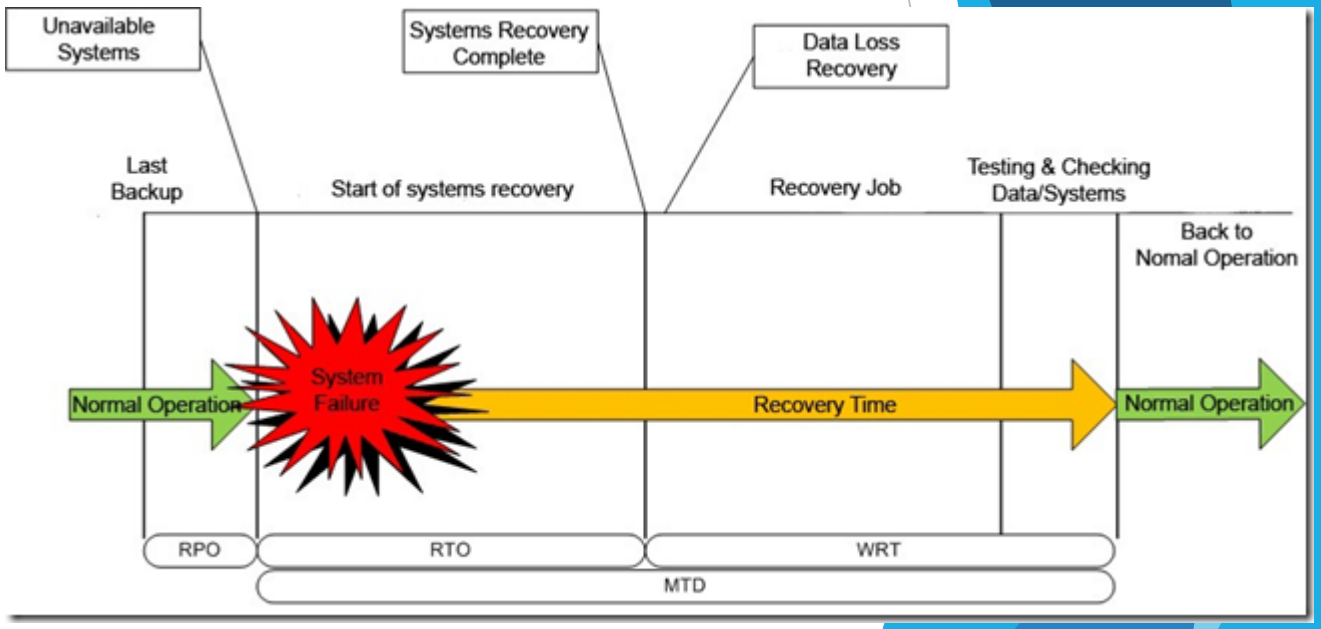
- Mesmo com recurso à redundância, note-se que podem existir componentes comuns ao sistema redundante
  - Exemplo: alimentação elétrica, sistema de controlo da redundância
- Estes componentes comuns designam-se por **SPOF** (*Single Point Of Failure*) e devem ser evitados
- Podem também existir falhas ligadas ou não a componentes físicos que implicam a indisponibilidade do sistema
  - Exemplo: vulnerabilidade no sistema operativo
- Um **CMF** (*Common-Mode Fault*) é uma falha que provoca a indisponibilidade de mais do que um componente do sistema redundante
  - Um SPOF é sempre um CMF mas o inverso não é verdade

Business Continuity Management

- Vejamos ainda com mais detalhe as componentes do MTTR para que cumpra o MTD pretendido
- A falha pode ser menor e facilmente resolvida ou mais complexa
- Importa por isso assegurar aspetos como as cópias de segurança ou equivalente, permitindo dessa forma a retoma da situação imediatamente anterior à ocorrência da falha
- Deve-se contudo notar que o MTTR só termina quando a operacionalidade está integralmente recuperada
  - Ou seja, o tempo de reposição das cópias de segurança ainda é parte integrante do MTTR

Business Continuity Management

- É mais apropriado e útil estabelecer objetivos quer do tempo médio de recuperação da funcionalidade quer do volume de perda de dados que é aceite
  - ***Recovery Time Objective*** (**RTO**) é o tempo médio de recuperação dos sistemas e infraestruturas
  - ***Recovery Point Objective*** (**RPO**) é o tempo máximo de perda de dados aceite
  - ***Work Recovery Time*** (**WRT**) é o tempo necessário para repor os dados e aplicações e testá-los



Fonte: CISSWhat? - A CISSP Review

Business Continuity Management

- Em teoria, é sempre possível arranjar uma estratégia para assegurar a continuidade de negócio
- Contudo, algumas dessas estratégias (e cujo risco de acordo com a matriz de risco identificada é considerável) podem implicar um custo desapropriado ao benefício obtido
- Torna-se por isso necessário analisar e calcular o custo da mitigação do risco face ao benefício obtido

Business Continuity Management

- Definamos um ***Asset Value*** (**AV**) que representa o valor do item em causa
- A exposição ao risco desse item é representada pelo ***Exposure Factor*** (**EF**)
- A ***Single Loss Expenditure*** (**SLE**) é obtida pela multiplicação destes dois fatores

$$SLE = AV \times EF$$

- Pode-se também obter ou calcular a probabilidade de ocorrência de um dano no bem ao longo de um ano, isto é, o ***Annualized Rate of Occurrence*** (**ARO**), o que permite calcular a ***Annualized Loss Expectancy*** (**ALE**)

$$ALE = SLE \times ARO = AV \times EF \times ARO$$

- O valor obtido deve ser comparado com o custo da ação necessária para o mitigar

Business Continuity Management

Caso 1

- Imaginemos os seguintes fatores
  - AV = 500€
  - EF = 20%
  - ARO = 3
  - SLE = 500 x 20% = 100€
  - ALE = SLE x ARO = 100 x 3 = 300€
  - A ação deve ser considerada se o seu custo for <= 300€

Caso 2

- Imaginemos os seguintes fatores
  - AV = 1000€
  - EF = 30%
  - ARO = 2
  - SLE = 1000 x 30% = 300€
  - ALE = SLE x ARO = 300 x 2 = 600€
  - A ação deve ser considerada se o seu custo for <= 600€

Business Continuity Management

- O cálculo do AV pode não ser sempre tão simples
  - Qual o valor dos dados?
- Para prever a perda de dados (o que influencia o cálculo do RPO) pode-se recorrer a diversas estratégias como por exemplo
  - *Mirroring* para local ou instalações remotas
  - Cópias de segurança
- Ambas as estratégias possuem benefícios e inconvenientes
- De notar que o RTO é ou pode ser independente da estratégia adotada
  - Mas o WRT é influenciado pela estratégia

Business Continuity Management

- *Mirroring* para local remoto
  - Pode ser síncrono ou assíncrono
  - Facilita a recuperação em caso de desastre
  - O RPO e o WRT são nulos (se for síncrono) ou muito próximo disso (se for assíncrono)
  - Implica a necessidade de garantir a confidencialidade e integridade dos dados, o que vai provocar por sua vez uma maior latência na rede com possível impacto na disponibilidade de operação (e inerentemente no SLA)

Business Continuity Management

- A cópia de segurança tem três estratégias possíveis (i)
  - Integral/Completa
    - Copia todos os dados
    - Implica maior tempo de cópia (o que prejudica o RPO)
    - Implica menor tempo de reposição (o que beneficia o WRT)
  - Incremental
    - Necessita sempre de uma cópia integral/completa prévia
    - Copia todos os dados que foram alterados desde a cópia incremental anterior (ou a integral se for a primeira)

Business Continuity Management

- A cópia de segurança tem três estratégias possíveis (ii)
  - Diferencial
    - Necessita sempre de uma cópia integral/completa prévia
    - Copia todos os dados que foram alterados desde a cópia integral anterior
- Qual a melhor estratégia tendo em mente o RPO, o RTO e o WRT?
  - Depende…
    - Do ambiente de execução…
    - Da possibilidade de manter ou não o sistema em funcionamento durante a execução da cópia…

Business Continuity Management

- Se a estratégia possível for a cópia de segurança, qualquer que seja a sua metodologia,
  - Deve ser sempre feita em duplicado
  - Um dos meios que contém a cópia deve ser guardado num local remoto
  - …
  - E não esquecer que aleatoriamente deve ser verificada a cópia e a capacidade de reposição dos dados!

Administração de Sistemas ASIST

Tópico 3

Gestão de acessos

Pinto Leite, Jorge (jpl@isep.ipp.pt)

Gestão de acessos

- ***Network Access Control*** (**NAC**) é um conceito de controlo de acesso a uma infraestrutura pela implementação de políticas de segurança
- Tem como objetivos
  - Mitigar e/ou prevenir ataques *zero-day*
  - Aplicar uma política de segurança ao longo da infraestrutura
  - Usar identidades para realizar o controlo de acesso
- Estes objetivos podem ser alcançados através de vários mecanismos (políticas de segurança com definição de controlo de segurança, filtragem, prevenção, entre outros)
- O NAC vai atuar como um sistema de deteção e resposta automática que pode reagir em tempo real para impedir ameaças antes que causem danos

Gestão de acessos

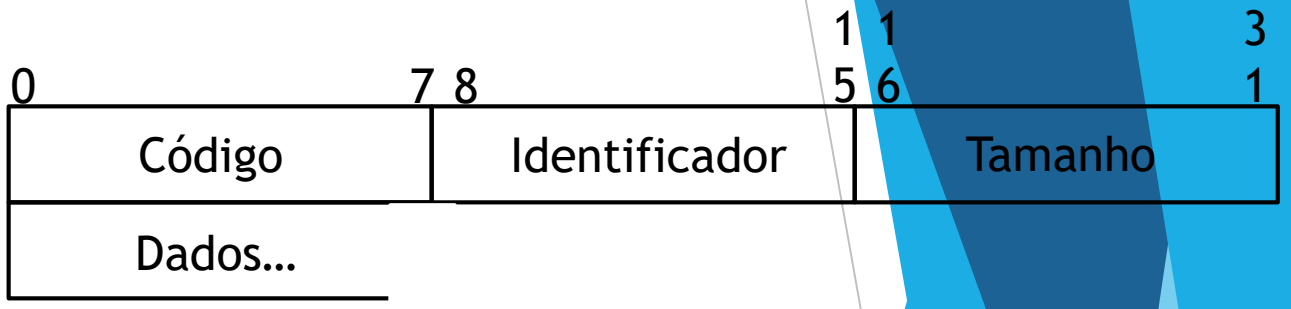
- O NAC pode ser implementado através de dois critérios ou mistura de ambos
  - Pré-admissão
    - Requer que um sistema cumpra todos os requisitos de segurança (tais como atualizações instaladas, antivírus atualizado) antes que possa comunicar com a rede
  - Pós-admissão
    - Permite ou nega acesso baseado na atividade do utilizador, que por sua vez é baseada numa matriz de autorizações previamente definida
- O critério *pré-admissão* refere-se a sistemas, serviços, e será objeto de análise mais tarde
- Já o critério *pós-admissão* é notório que envolve os utilizadores que acedem à rede
- E como é realizada a identificação dos utilizadores?

Gestão de acessos

- Os utilizadores representam os intervenientes de uma infraestrutura informática
- Necessitam de autenticação na infraestrutura para poderem tirar partido dos seus recursos
- A autenticação mais simples é denominada ***Password Authentication Protocol*** (**PAP**) e consiste na troca de duas mensagens no início do estabelecimento da autenticação
  - O utilizador envia a identificação e a senha
  - O sistema de autenticação valida o par identificação/senha
  - Devolve a autorização ou negação de acesso

Gestão de acessos

- Este método é considerado inseguro porque
  - A identificação e a senha são enviadas em texto claro (legível)
  - Não há proteção contra repetições ou falhas sucessivas
  - A identificação e a senha necessitam ser previamente definidas (segredo partilhado)
- O código representa a operação em curso
  - 1 – Authentication request
  - 2 – Authentication Ack
  - 3 – Authentication Nak



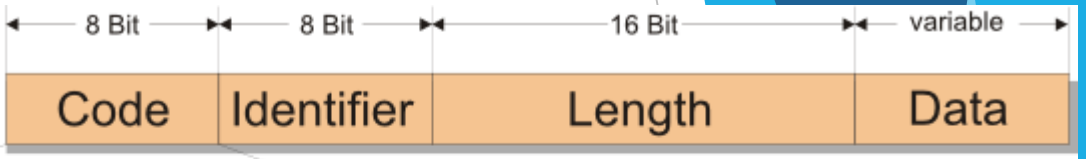
Constituição do campo *dados*

Tamanho da ID	Identificação
Tamanho da senha	Senha

Gestão de acessos

- Um método alternativo e mais seguro é aplicado pelo ***Challenge-Handshake Authentication Protocol*** (CHAP)
- Este protocolo utiliza um sistema de três mensagens e pode verificar periodicamente a identidade do utilizador
- Continua porém a ser necessária a definição prévia da identificação e da senha (segredo partilhado)
- O campo Código indica a operação em curso
- O campo Identificador é um número aleatório e serve para mapear os desafios e as respostas
- O campo Dados contém
  - O tamanho do texto de desafio ou resposta
  - O desafio (se código = 1) ou a resposta (se código = 2)
  - Um nome que identifica a o dispositivo que enviou o desafio

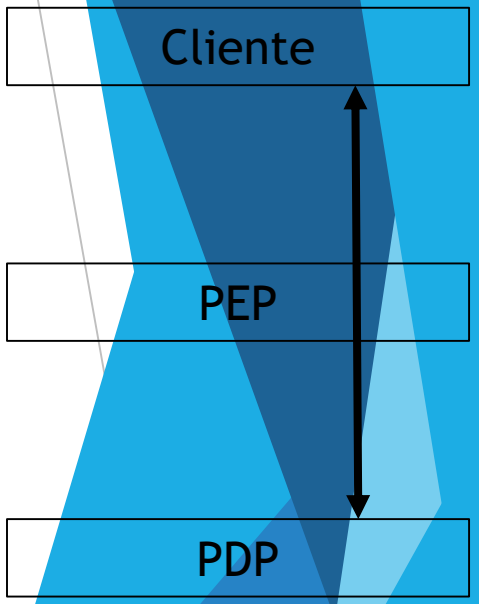
O desafio é um número aleatório gerado pelo serviço e ligado à senha
Varia de sessão para sessão



Coding (hex)	Meaning
01	Challenge
02	Response
03	Success
04	Failure

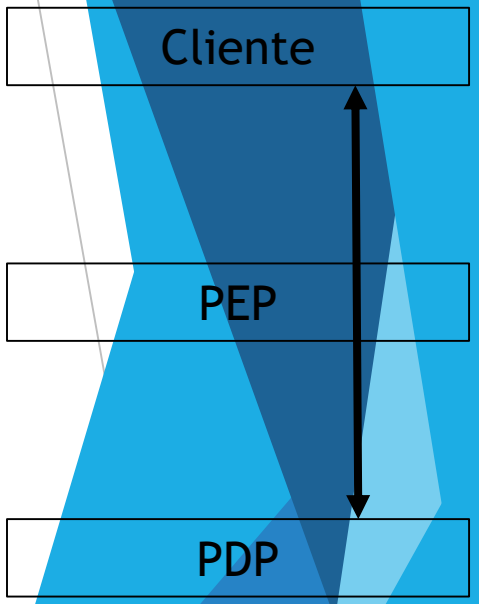
Gestão de acessos

- O ponto de acesso à rede é normalmente designado por ***Policy Enforcement Point*** (**PEP**) que, por sua vez, deverá:
  - Ou conter a identificação e credenciais dos utilizadores para validar o acesso
  - Ou contactar um ***Policy Decision Point*** (**PDP**) que se encarregará de validar as credenciais inseridas



Gestão de acessos

- Apesar de não ser especificamente um protocolo, convém olhar também para o ***Extensible Authentication Protocol*** (**EAP**)
- Na realidade é uma estrutura (*framework*) para autenticação
- Permite múltiplos sistemas alternativos na autenticação, como o uso de cartões, biometria, etc.
- Permite que o PEP funcione em modo *pass-through*, por exemplo quando a conexão é efetuada através de um ponto de acesso sem fios
- Na prática possibilita que o PEP seja transparente e a autenticação seja diretamente no PDP
- Vários mecanismos de segurança são possíveis de implementar recorrendo ao EAP, dos quais se destaca o ***Protected Extensible Authentication Protocol*** (**PEAP**) que encapsula o EAP num túnel ***Transport Layer Security*** (**TLS**)
- As normas ***Wifi Protected Access*** (**WPA**) incluem PEAP



Gestão de acessos

- Um dos mecanismos de autenticação mais utilizados baseia-se num segredo partilhado (como por exemplo uma senha de acesso)
- O segredo pré-partilhado enquadra-se na cifra simétrica, que veremos noutro tópico
- O segredo pré-partilhado garante a confidencialidade já que exige a chave para decifrar a mensagem
- Alguns autores consideram que este método não garante a autenticidade
  - Porque o segredo pode ser fraco ou obtido por engenharia social (à qual voltaremos noutro tópico)
- Para um segredo pré-partilhado ser efetivo, deve ser exclusivo de um par de entidades
  - Ou seja, para N entidades deverão existir $\mathbf{C}_{(n,2)}$ (combinações de N dois a dois)

Gestão de acessos

- O elevado número de segredos conduz a problemas humanos
  - Como geramos e nos recordamos de um número enorme de segredos?
- Implica também a criação prévia do segredo – por exemplo, uma conta de utilizador
- Na prática, isto significa que uma entidade anteriormente desconhecida nunca se poderá autenticar
- Também num contexto prático, como partilhamos o segredo com uma entidade? Envio por SMS?

Gestão de acessos

- A autenticação das entidades insere-se na gestão de identidades ***Identity Management***
- Pode ser centralizada ou descentralizada
  - A autenticação descentralizada (também conhecida por controlo de acesso distribuído) implica que várias entidades localizadas ao longo do sistema ou infraestrutura executem a verificação do acesso
    - A sincronização é complexa e exige um elevado esforço administrativo
    - Os utilizadores têm a tendência de usar a mesma senha em todos os serviços e sistemas, ou a anotar as senhas em papel, com implicação direta na segurança
    - Contudo oferece vantagens no controlo de autorização de acessos
  - A autenticação centralizada exige menor esforço administrativo, não implica a sincronização das senhas, e as entidades necessitam de se autenticar apenas uma vez
    - Em suma, oferece o ***Single Sign-On*** (**SSO**)
    - Mas obriga a um esforço acrescido de autorização de acessos

Gestão de acessos

- Nas organizações é especialmente sentida a necessidade de assegurar a gestão centralizada dos utilizadores
- Um serviço de diretório consiste numa base de dados que inclui informação sobre objetos e entidades
- Vários desses serviços são baseados no ***Lightweight Directory Access Protocol*** (**LDAP**) RFC 3494
  - O Active Directory Domain Services da Microsoft, por exemplo, é baseado no LDAP
- O LDAP pode ser considerado uma evolução do protocolo X.500, suportando o diálogo com ele e mantendo várias das suas definições (como classes e atributos)
- Uma vantagem importante do LDAP é suportar a exportação de ramos da árvore (ou da árvore toda, claro!)

Gestão de acessos

- O LDAP (RFC 4511 na sua versão mais atual) é uma base de dados não relacional que pode alojar as autenticações das entidades
- A sua estrutura de dados é muito flexível e otimizada para operações de consulta
- A sua estrutura é em árvore, sendo cada ramo um **objeto** que possui **atributos**
- O topo da estrutura é designado por **root**
- Os atributos são indexados com o intuito de otimizar a pesquisa/consulta
- O cliente (humano ou sistema) acede ao diretório global (que pode conter vários servidores LDAP) através de um cliente ou ***Directory User Agent*** (**DUA**)
- Este cliente por sua vez interage com um ou mais servidores ou ***Directory System Agents*** (**DSA**) através do protocolo LDAP

Gestão de acessos

- O protocolo LDAP utiliza o TCP como mecanismo de transporte e baseia-se na ***Abstract Syntax Notation One*** (**ASN.1**)
- Todas as mensagens LDAP são encapsuladas num formato denominado ***LDAPMessage***, sequência em formato string que contém um número de sequência, um identificador, a operação pretendida e uma mensagem
- Caso uma mensagem em que o número de sequência não seja reconhecido, a conexão é terminada devolvendo uma indicação de desconexão com código de erro (**ProtocolError**) ao cliente
- Em todos os outros casos em que não seja possível ao cliente ou ao servidor analisar a mensagem, a conexão é de imediato terminada podendo ou não devolver uma mensagem ao outro extremo

Gestão de acessos

- A definição dos atributos dos objetos no LDAP tem que cumprir a definição do *schema* (definidos na RFC 4512) não sendo portanto livre
- O *schema* contém as classes possíveis contendo cada uma os atributos que são válidos e que tipo de utilização é permitida em cada um
  - Alguns atributos são obrigatórios, outros opcionais
- A definição das classes implementa o conceito de herança, pelo que uma classe ao ser subclasse de outra incorpora os atributos da hierarquicamente superiora, podendo na subclasse ser definidos mais atributos

Gestão de acessos

- Existem classes estruturais e auxiliares (contudo, opcionais)
- Cada objeto pertence a uma única classe estrutural, mas pode pertencer adicionalmente a várias classes auxiliares
- Entre os atributos obrigatórios encontra-se o ***Domain Component*** (**DC**) que pode pertencer à classe ***dcObject*** (que é auxiliar) normalmente conjugada com a classe estrutural ***Organization***
  - Alternativamente poder-se-á usar a classe estrutural **Domain**
- Tornou-se frequente a nomeação baseada no nome de domínio DNS, sendo cada uma das suas partes um **dc**, separados por vírgulas
  - DNS: **isep.ipp.pt**
  - root: **dc=isep,dc=ipp,dc=pt**

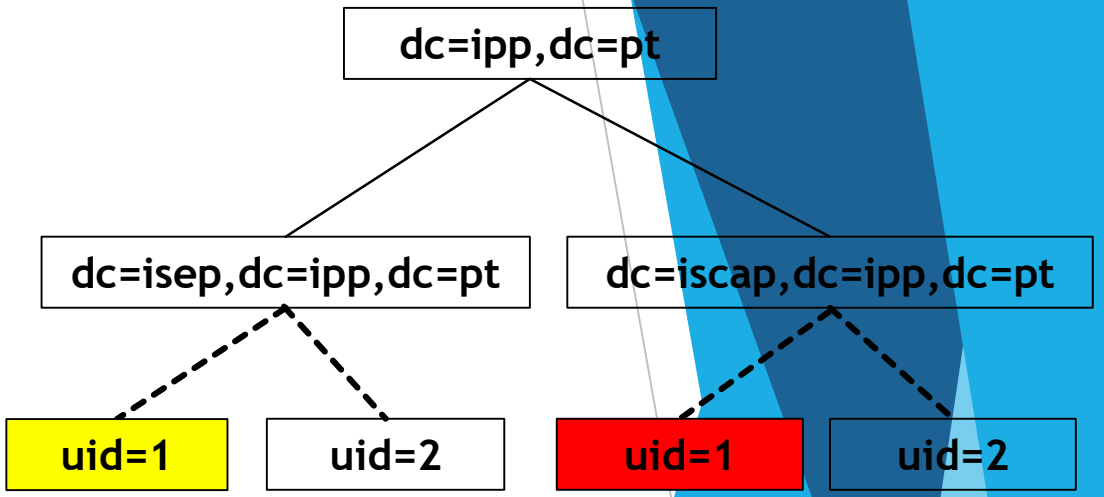
Gestão de acessos

- O nome que identifica um objeto único e exclusivo na árvore LDAP é um ***Distinguished Name*** (**DN**)
- Se o DN corresponde ao ***root***, designa-se por **base DN** ou **root DN**
- Um DN tem de ser único no LDAP identificando de forma inequívoca o objeto
- Os nomes dos atributos contudo podem ser repetidos ao longo da estrutura, desde que em ramos diferentes da árvore
- Distingue-se então os ***Relative Distinguished Names*** (**RDN**) que, sendo únicos e exclusivos no ramo em que se encontram, podem ser repetidos noutro ramo
- Na imagem, os objetos com RDN **1** têm o DN

  (com fundo amarelo) **uid=1,dc=isep,dc=ipp,dc=pt**

  (com fundo vermelho) **uid=1,dc=iscap,dc=ipp,dc=pt**

Sob dc=isep, dc=ipp, dc=pt só pode existir um uid=1, sendo um RDN, mas podem existir outros uid <> 1, tal como sob dc=iscap, dc=ipp, dc=pt

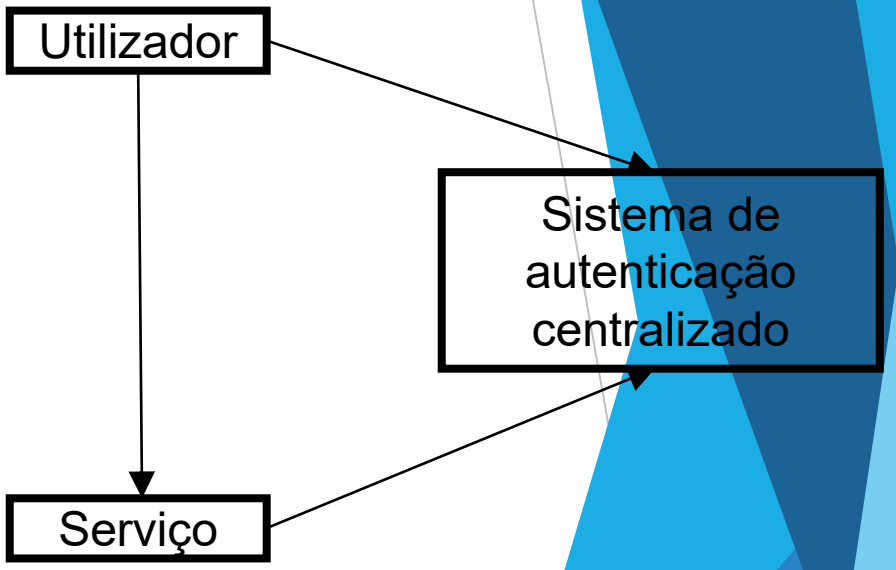


Gestão de acessos

- Quando uma entidade se autentica numa infraestrutura há uma sequência de passos necessários
- Em primeiro lugar, a **identificação**
  - Consiste em anunciar quem é a entidade (por exemplo, o nome de utilizador, ao qual chamaremos ID, e a senha de acesso)
- Depois, a **autenticação**
  - O sistema valida o par ID – senha
  - Devolve o sucesso ou insucesso da identificação
- Estes passos terão de ser efetuados para todos os serviços a quem o ID pretende aceder
- Mas ter-lhes-á acesso (autorização)?

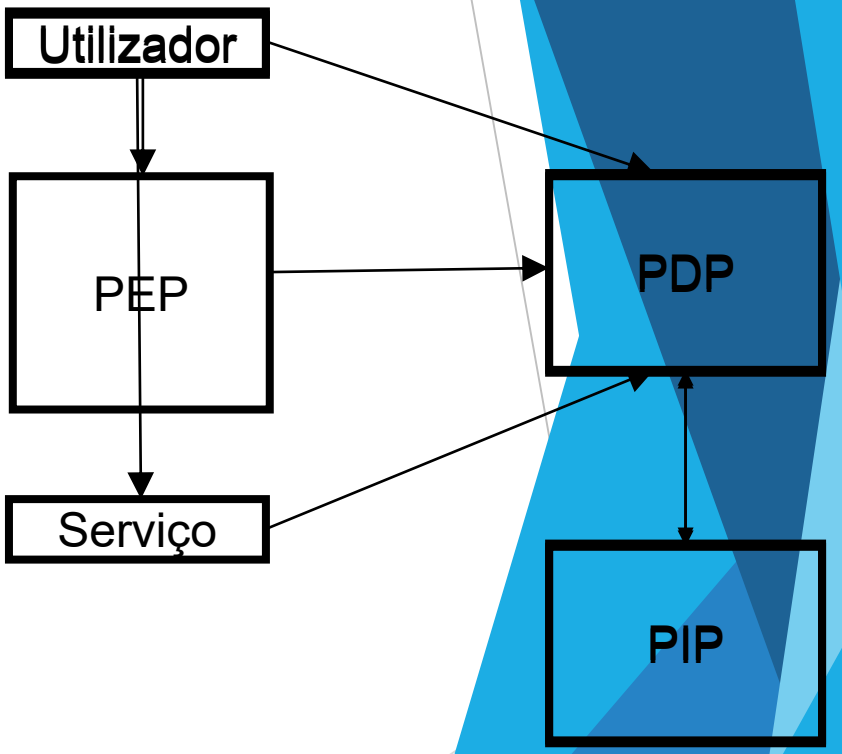
Gestão de acessos

- A **autenticação** refere-se a comprovar a identificação de quem se autentica
- A **autorização** refere-se a verificar se o ID pode aceder ao serviço
  - E, claro, só ocorre a quem já se autenticou com sucesso
- Torna-se assim clara a vantagem de ter um serviço centralizado de autenticação, já que permite que todos os serviços validem o utilizador
- Isto não implica porém que o utilizador tem autorização garantida a todos os serviços; implica é que o serviço pode validar o utilizador recorrendo ao serviço centralizado de autenticação



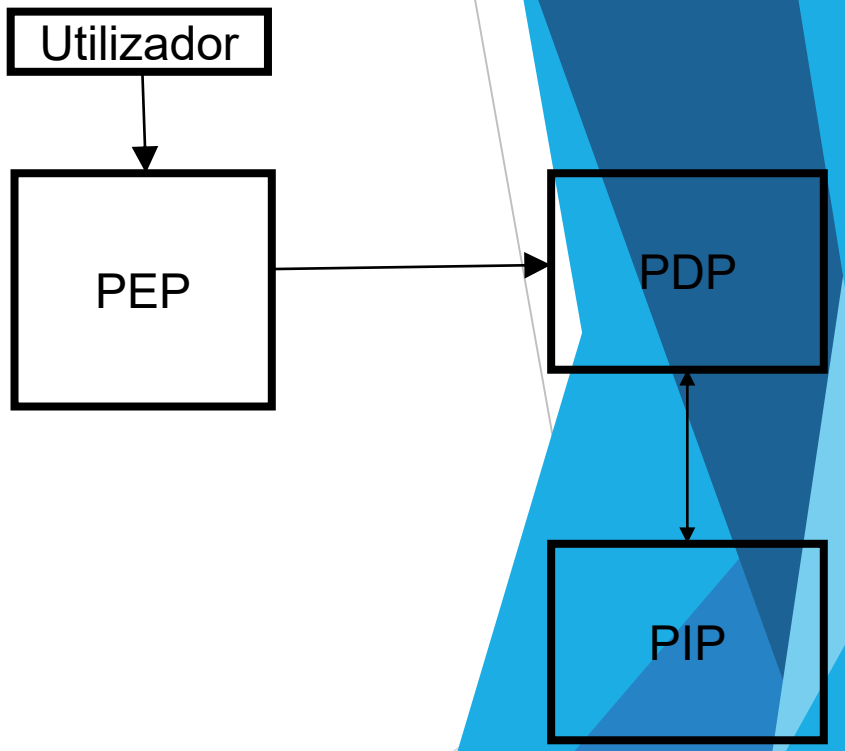
Gestão de acessos

- O serviço central de autenticação é habitualmente designado por ***Policy Decision Point*** (**PDP**)
- A validação da autorização concedida ao utilizador recorre a outro serviço centralizado, intimamente ligado ao PDP, designado por ***Policy Information Point*** (**PIP**)
- O utilizador normalmente não acede diretamente ao PDP mas a um serviço que se encarrega de contactar com o PDP
- A este serviço chama-se ***Policy Enforcement Point*** (**PEP**)
- O PEP é normalmente um serviço de rede, mas pode também ser um acesso por túnel VPN (a que voltaremos noutro módulo) ou um Access Point
  - Nesse caso denomina-se habitualmente por ***Network Access Service*** ou ***Network Access Server*** (**NAS**)

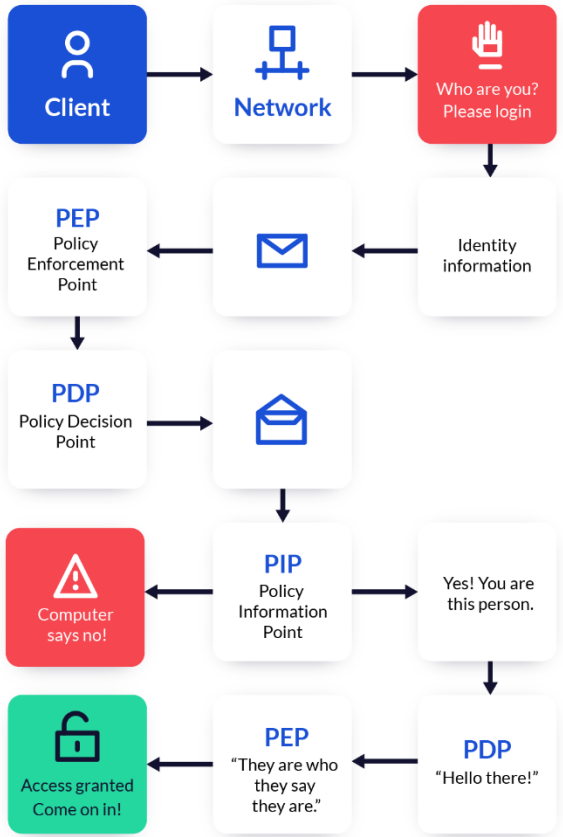


Gestão de acessos

- O serviço central de autenticação é habitualmente designado por ***Policy Decision Point*** (**PDP**)
- A validação da autorização concedida ao utilizador recorre a outro serviço centralizado, intimamente ligado ao PDP, designado por ***Policy Information Point*** (**PIP**)
- O utilizador normalmente não acede diretamente ao PDP mas a um serviço que se encarrega de contactar com o PDP
- A este serviço chama-se ***Policy Enforcement Point*** (**PEP**)
- O PEP é normalmente um serviço de rede, mas pode também ser um acesso por túnel VPN (a que voltaremos noutro módulo) ou um Access Point
  - Nesse caso denomina-se habitualmente por ***Network Access Service*** ou ***Network Access Server*** (**NAS**)



Gestão de acessos

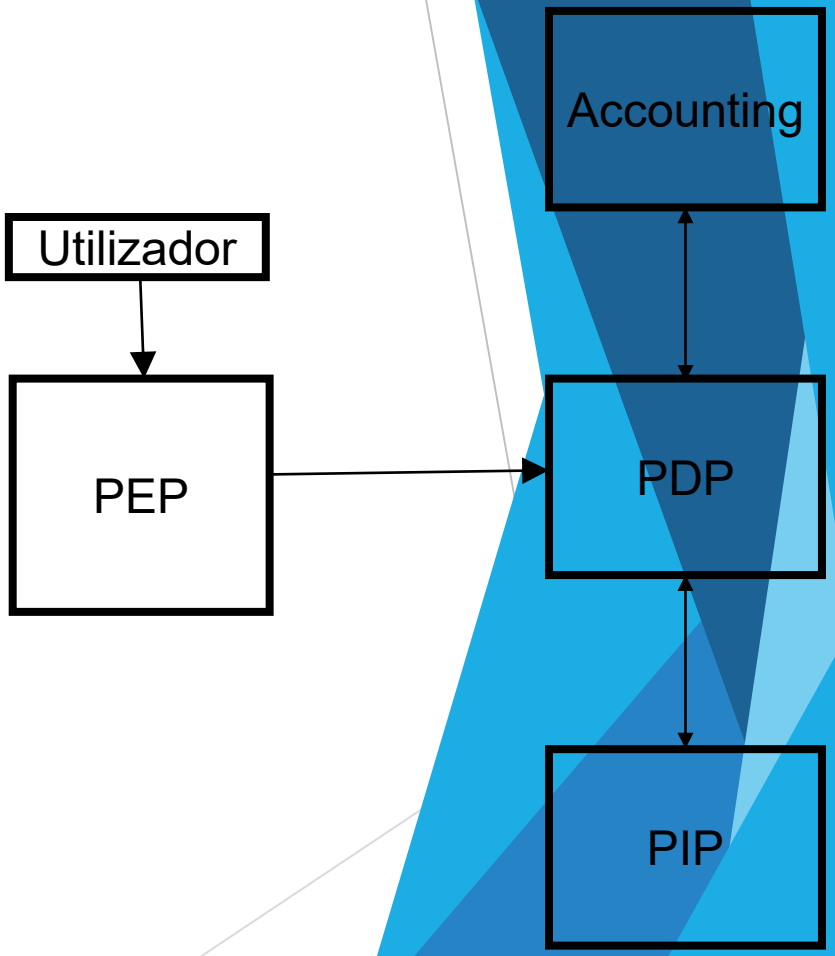


CODEBOTS
Authorised reseller

Fonte: codebots.com

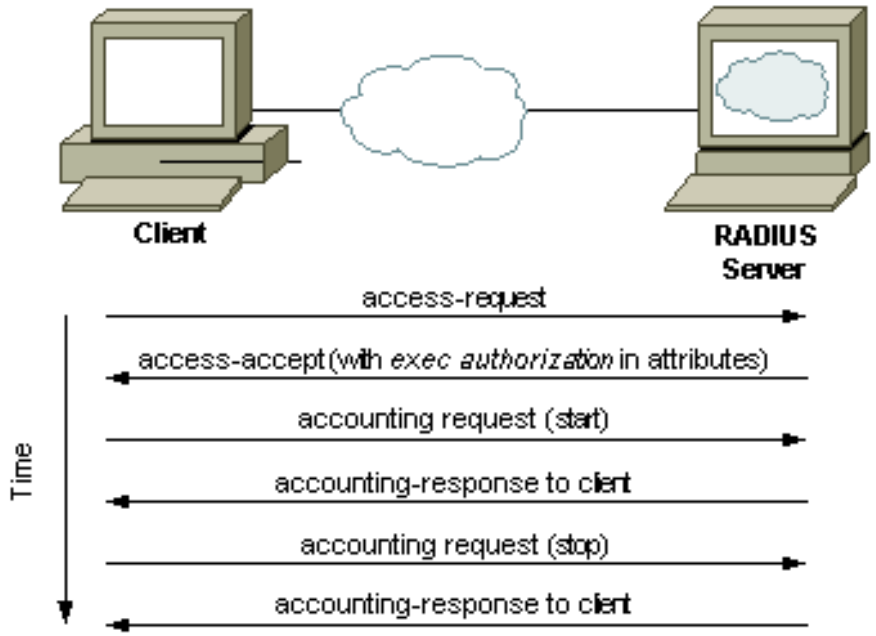
Gestão de acessos

- É normalmente necessário (seja para controlo interno seja para cumprir obrigações legais) o registo dos acessos concedidos e negados às entidades
- Este registo denomina-se ***Accounting*** e, juntamente com a ***Authentication*** e ***Authorization*** anteriores, constitui os sistemas **AAA**
  - *AAA - Authentication, Authorization, Accounting*

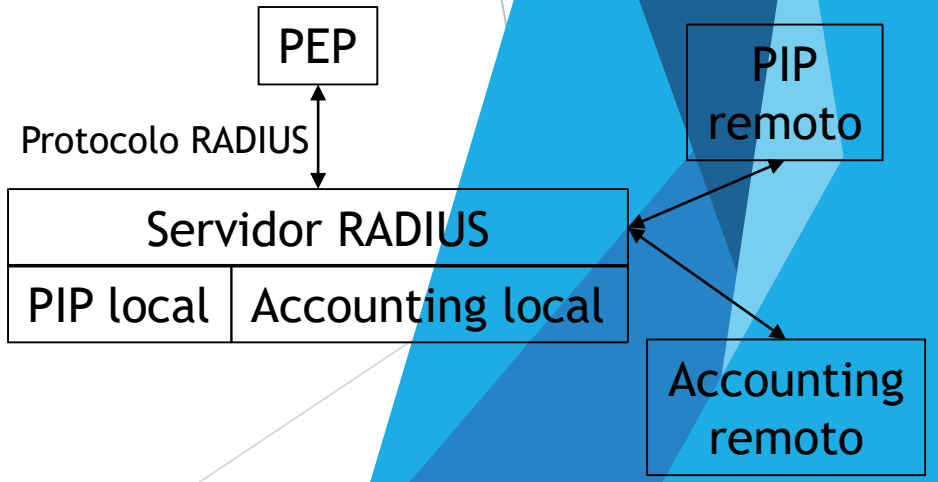


Gestão de acessos

- Foram desenvolvidos protocolos para assegurar o suporte ao AAA
- Entre estes, destaca-se o ***Remote Authentication Dial In User Service*** (**RADIUS**) (**RFC 2865**)
- O RADIUS é em simultâneo um serviço e um protocolo que assegura as comunicações entre o PEP e o PDP
- Esta comunicação ocorre em UDP para evitar constrangimentos associados ao TCP
- Pode ter um PIP e accounting local ou, através de extensões, comunicar com outros serviços que implementem essas funções
- No RADIUS os pedidos de accounting são registados apenas para o início e fim da sessão
- Usa 8 bits para definir os pares Atributo – Valor (AV)

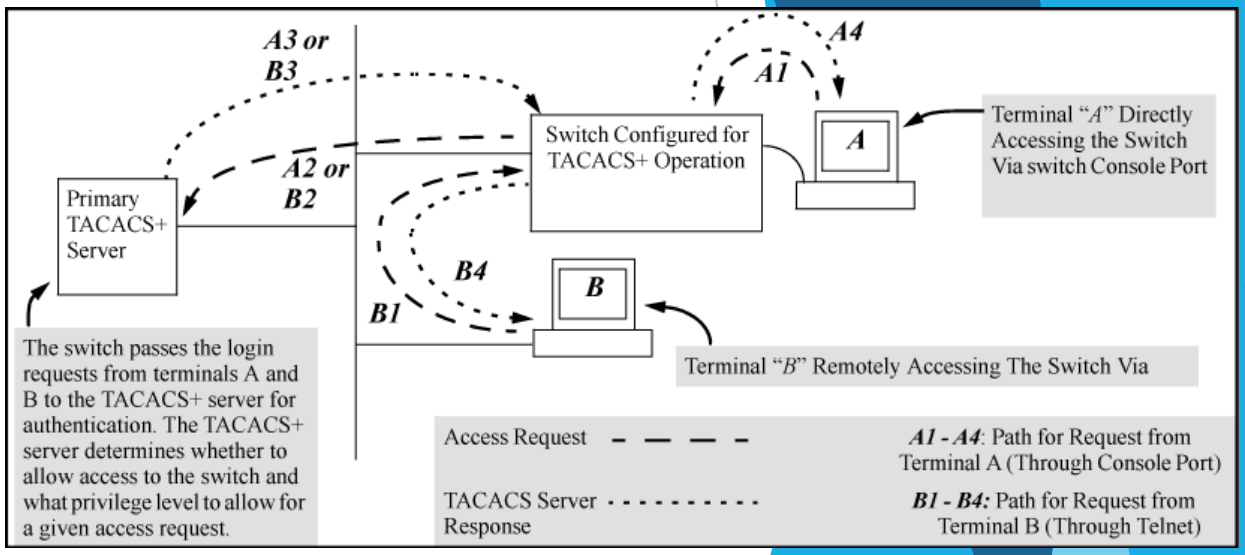


Fonte: cisco.com



Gestão de acessos

- Outro protocolo com suporta o AAA é o ***Terminal Access Controller Access-Control System Plus*** (**TACACS+**)
  - Evolução do TACACS e do XTACACS mas incompatíveis com eles
- Ao contrário do RADIUS usa TCP
- Uma diferença significativa é que o TACACS+ isola as verificações (autenticação e autorização) e repete a autorização sempre que a execução de um novo comando é requerida



Fonte: techhub.hpe.com

Gestão de acessos

- Mais recentemente, em 2003, surgiu uma evolução baseada no RADIUS embora não compatível com ele: o ***Diameter*** (RFC 6733 na versão atual)
- Como aspetos mais importantes,
  - Suporta o EAP
  - Suporta **SCTP** (***Stream Control Transmission Protocol***)
  - Permite mecanismos de *failover*
  - É baseado em TCP
  - Usa 32 bits para definir os pares Atributo – Valor (***Attribute-Value Pair*** **AVP**) sendo por isso mais escalável
    - Os AVP contêm informação de autenticação, accounting, autorização, encaminhamento, segurança e configuração para o pedido e resposta
  - Suporta redes cabladas, wifi, 3G, IP Multimedia Systems (IMS), e LTE/4G

Gestão de acessos

- Um método usado para suportar o AAA consiste em ter uma terceira parte que assegure a identificação e autenticação, fornecendo bilhetes (*tickets*) às entidades que os solicitem
  - Oferecendo o que é designado como ***ticket authentication***
- O sistema mais comum deste tipo é o ***Kerberos***

Gestão de acessos

- O **Kerberos** foi desenvolvido pelo MIT para uso interno, sendo mais tarde disponibilizado para uso externo
- A versão mais atual é a V5, descrita na RFC 4120
- Oferece uma solução de ***Single Sign-On*** (**SSO**) aos utilizadores e providencia proteção às credenciais de autenticação
- Foi desenhado para assegurar uma forte autenticação para aplicações cliente/servidor através do uso de criptografia de chave secreta
- Através de extensões descritas na RFC 4556 a autenticação inicial pode socorrer-se da ***Public Key Infrastructure*** (**PKI**), à qual voltaremos mais tarde

Gestão de acessos

- Cada uma das entidades a quem o Kerberos pode distribuir chaves designa-se por ***principal***
- O Kerberos divide a infraestrutura em ***realms***
- Cada *realm* contém obrigatoriamente o serviço de distribuição de chaves denominada ***Key Distribution Center*** (**KDC**)
- Todos os utilizadores e recursos (*principals*) estão registados no KDC, que mantém uma base de dados das chaves de todos
- O servidor de autenticação executa as funções do KDC, dois serviços distintos e paralelos
  - ***Authentication Service*** (**AS**)
  - ***Ticket Granting Service*** (**TGS**)

Gestão de acessos

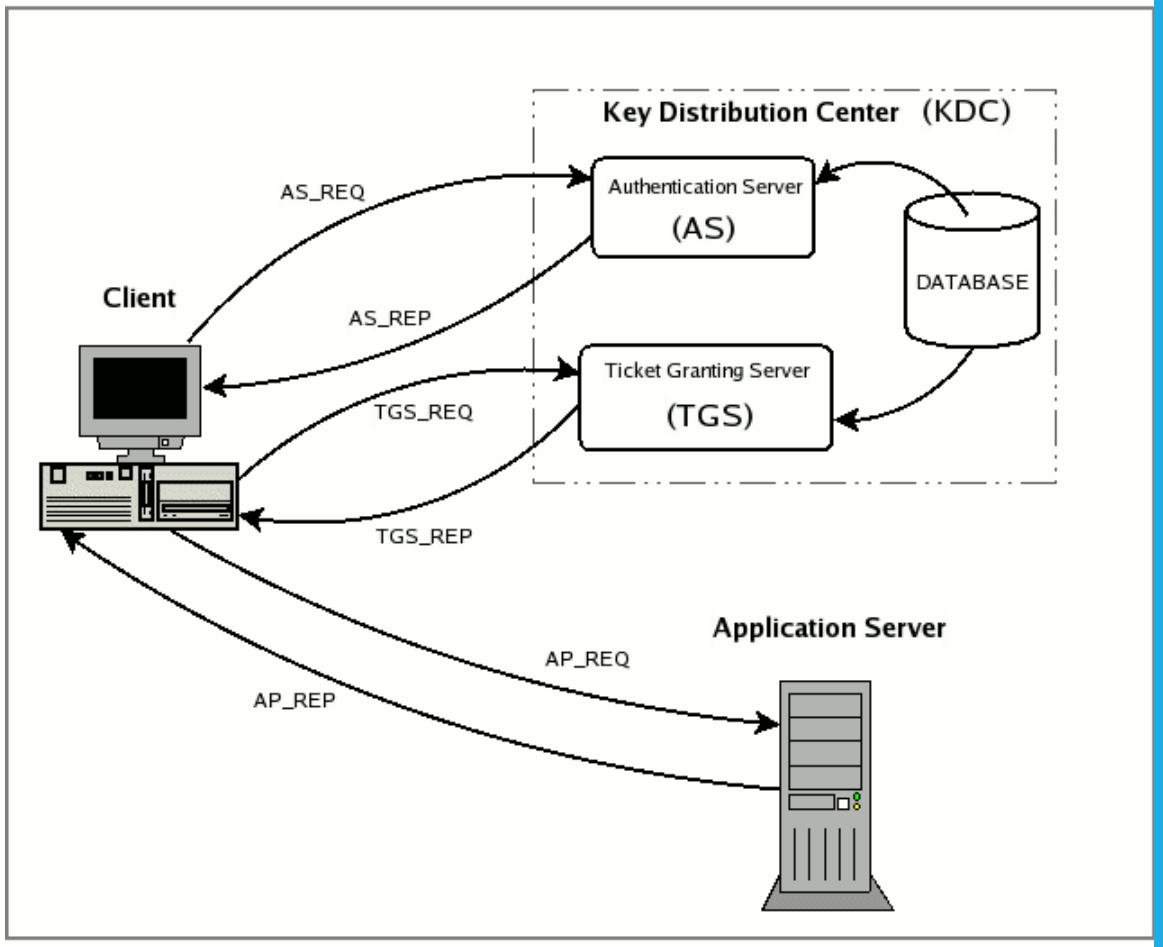
- ***Authentication Server*** **(AS)**
  - Possui uma chave partilhada com os *principals*
  - Aceita ou rejeita a tentativa de acesso
  - Fornece aos *principals* um *ticket* que lhes permite provar a sua identidade ao TGS
- ***Ticket Granting Service*** **(TGS)**
  - Fornece os ***Ticket-Granting tickets*** **(TGT)**, chaves de sessão para comunicação temporárias, aos *principals*
  - Temporárias, pois o TGT tem um tempo de vida
  - Se não houver sincronização horária nos *principals*, o acesso será negado
  - O TGT prova que o *principal* se autenticou no KDC e está autorizado a pedir acessos aos recursos

Gestão de acessos

- Os *tickets* são mensagens encriptadas que provam a autorização de acesso a um recurso
- Um utilizador pede um *ticket* para aceder a um recurso, e se ambos (utilizador e recurso) estão autenticados e o utilizador tem permissão para utilizar o recurso, o Kerberos (mais exatamente o TGS) fornece-o
- Para além do tempo de vida, os *tickets* Kerberos possuem parâmetros de utilização específicos
- Quando um *ticket* expira o cliente deve solicitar a sua renovação ou um novo *ticket* para continuar a usar o recurso
- Estando todos os *principals* autenticados, assegura a toda a estrutura a identidade e autorização uns dos outros

Gestão de acessos

- Quando um *principal* se liga ao *realm* envia um pedido de autenticação ao KDC
- Sendo válida, é-lhe devolvido um Ticket-Granting Ticket (TGT)
- Quando o *principal* acede após a autenticação a um recurso da rede o TGT que recebeu é usado para pedir um outro *ticket* de acesso ao recurso
- Este novo pedido é enviado ao KDC que o valida

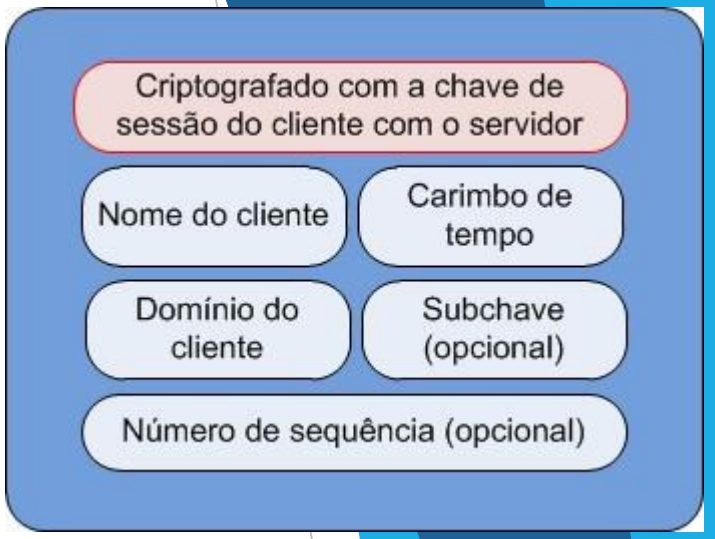


Fonte: kerberos.org

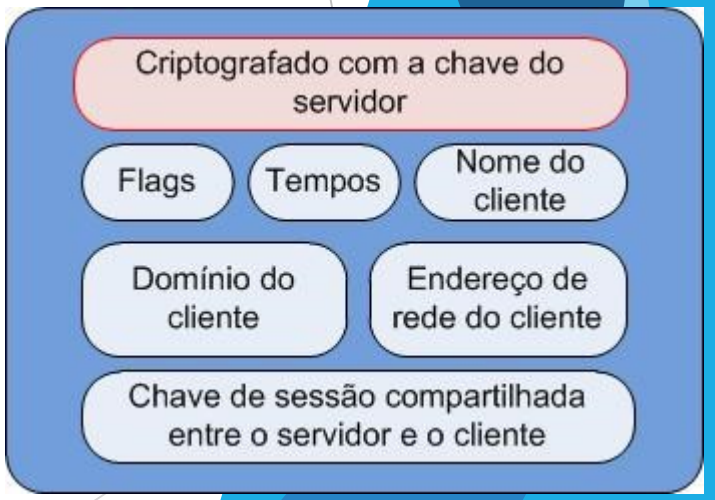
Gestão de acessos

- O processo de acesso no Kerberos tem o seguinte funcionamento
  1. O utilizador insere a autenticação no cliente
  2. O cliente encripta a autenticação e envia-a ao KDC
  3. O KDC verifica a autenticação na base de dados do AS
  4. O KDC gera uma chave simétrica para ser usada na comunicação; encripta-a com o *hash* da senha do utilizador; gera também um TGT encriptado
  5. O KDC transmite a chave simétrica e o TGT (ambos encriptados) ao cliente
  6. O cliente usa o TGT até expirar e decifra a chave simétrica com o seu *hash*

Autenticação



Ticket

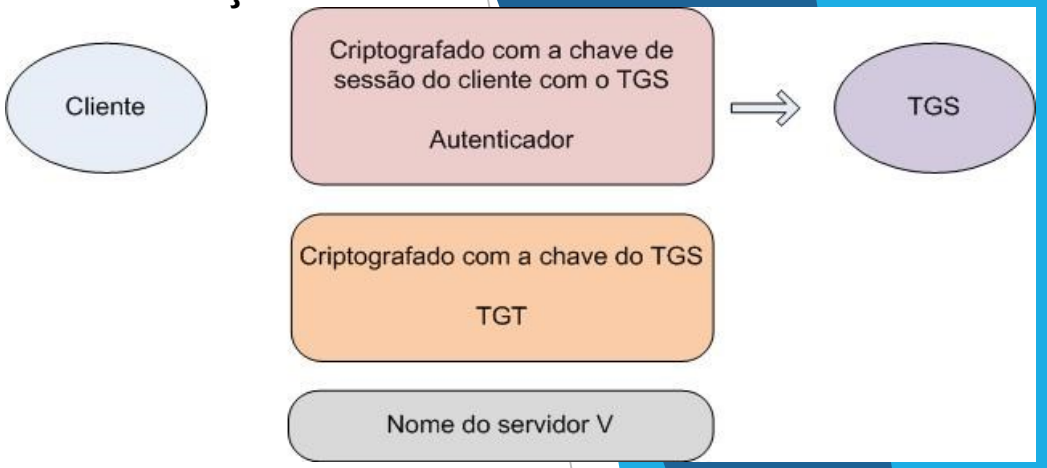


Fonte: UFRJ

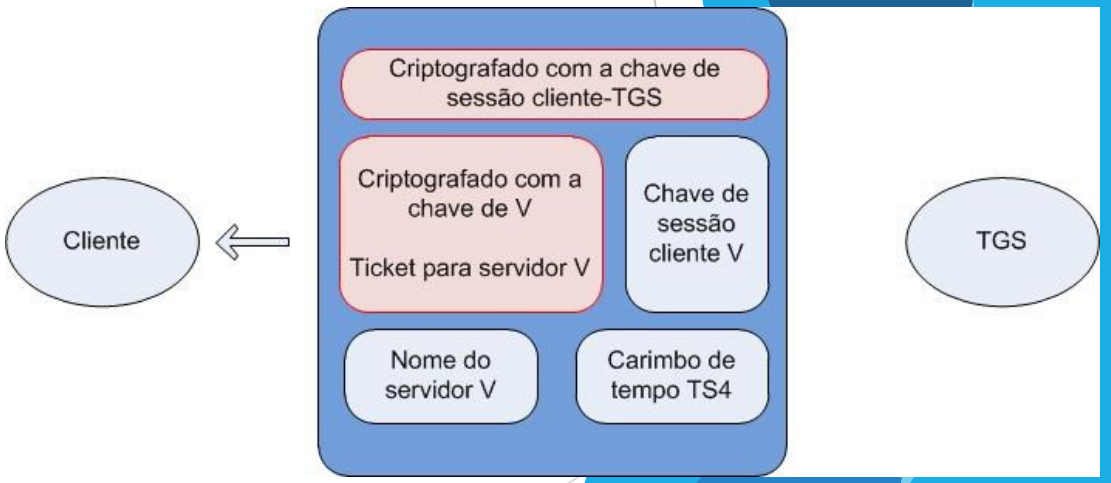
Gestão de acessos

- Quando um cliente quer aceder a um recurso, a sequência é
  1. O cliente envia o seu TGT ao KDC junto com o pedido de acesso ao recurso
  2. O KDC verifica a validade do TGT e se o cliente tem autorização para usar esse recurso
  3. O KDC gera um ticket de acesso ao recurso e envia-o ao cliente
  4. O cliente envia o ticket de acesso ao recurso ao servidor ou serviço onde o recurso está alojado
  5. O servidor ou serviço verifica a validade do ticket com o KDC
  6. Após verificação da identidade e autorização do acesso, a atividade do Kerberos está concluída
  7. O servidor ou serviço que aloja o recurso abre uma sessão com o cliente e a comunicação é estabelecida

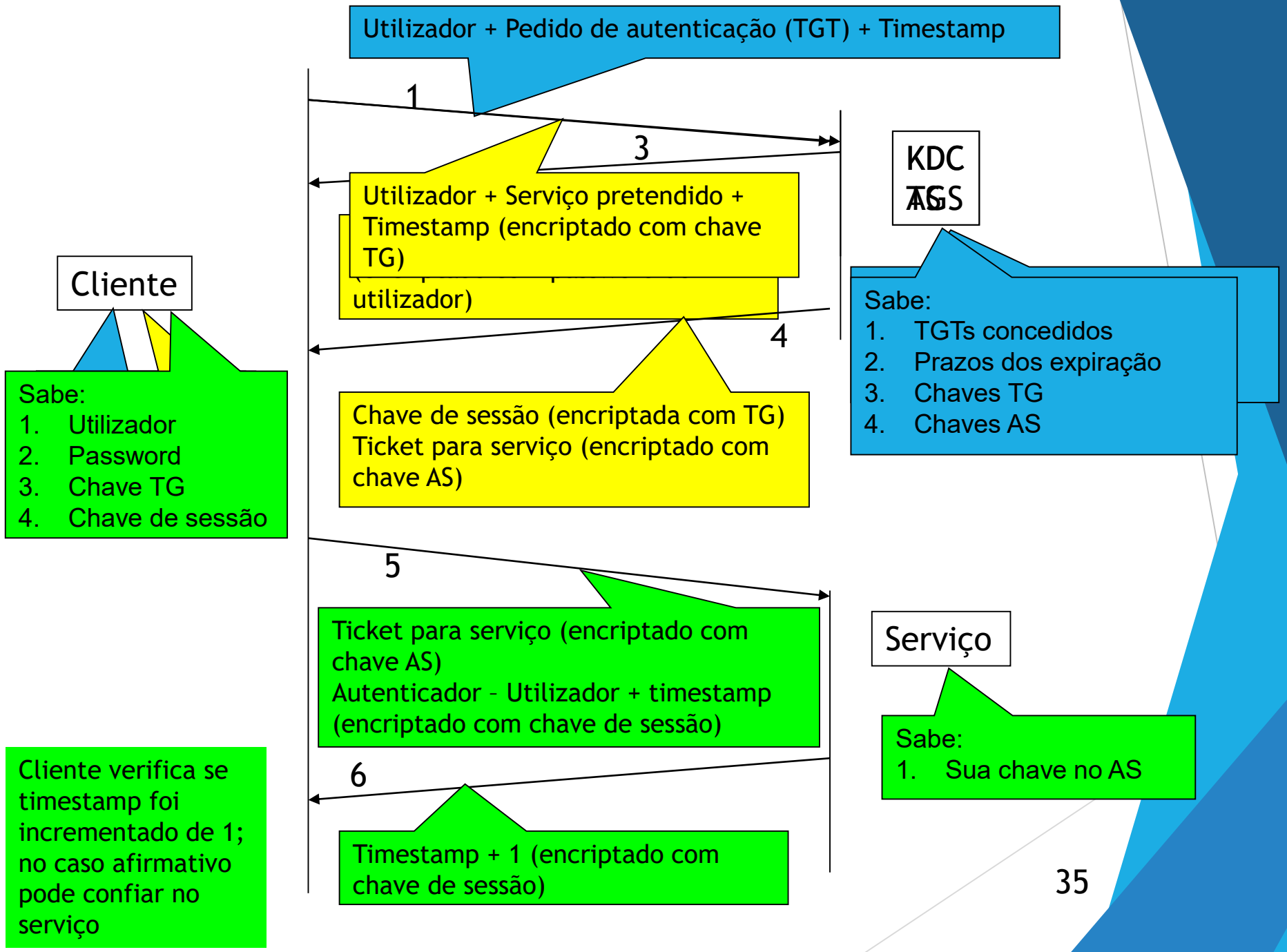
Solicitação de *ticket* de acesso a recurso



***Ticket* de acesso ao recurso**



Fonte: UFRJ


Utilizador + Pedido de autenticação (TGT) + Timestamp
1
3
KDC
TGS
AS
Utilizador + Serviço pretendido + Timestamp (encriptado com chave TG)
utilizador)
Cliente
4
Sabe:
1. TGTs concedidos
2. Prazos dos expiração
3. Chaves TG
4. Chaves AS
Sabe:
1. Utilizador
2. Password
3. Chave TG
4. Chave de sessão
Chave de sessão (encriptada com TG)
Ticket para serviço (encriptado com chave AS)
5
Ticket para serviço (encriptado com chave AS)
Autenticador - Utilizador + timestamp (encriptado com chave de sessão)
Serviço
Sabe:
1. Sua chave no AS
Cliente verifica se timestamp foi incrementado de 1; no caso afirmativo pode confiar no serviço
6
Timestamp + 1 (encriptado com chave de sessão)

Gestão de acessos

- Os nomes dos *principals* num *realm* possuem reservas
  - O nome ***krbgt***, por exemplo, é reservado
  - Os símbolos **@** e **/** são usados para separar o nome do *principal* do nome do *realm* e componentes do nome do *principal* respetivamente
  - Exemplos são **user@realm** e **sftp/server@realm**

Gestão de acessos

- O Kerberos pode gerir relações de confiança entre *realms* distintos
- Com esta relação de confiança é possível a um *principal* de um *realm* aceder a recursos de um *realm* distinto sem necessitar de possuir credenciais no AS desse outro *realm*
- Este processo consiste em definir cada TGS como *principal* no outro *realm*
- Independentemente do *realm* em que se encontra, cada *principal* pode obter um *ticket* de acesso a recursos do outro *realm*
- Considera-se que um *realm* pode comunicar com outro *realm* se ambos partilham uma chave *inter-realm*
  - Esta chave será usada para encriptar os *tickets* de pedido de acesso a recursos do *realm* remoto

Gestão de acessos

- O Kerberos suporta também hierarquia entre *realms*
  - Um exemplo são os domínios Active Directory da Microsoft na mesma *tree*
- Havendo hierarquia, a relação de confiança é transitiva, isto é, se o *realm* A confia no *realm* B e o *realm* B confia no *realm* C, então o *realm* A confia no *realm* C

Gestão de acessos

- Como desvantagem do Kerberos, notar que o KDC é um SPOF
- As chaves baseiam-se nas senhas dos utilizadores, pelo que uma senha "fraca" potencia a sua captura e posterior acesso à chave
- O Kerberos não disponibiliza mecanismos de tolerância a falhas, pelo que é razoável configurar mais do que um KDC no mesmo *realm*, contudo:
  - A sincronização das bases de dados tem de ser feita com recurso a sistemas externos (automáticos ou manuais)
  - As aplicações deverão estar preparadas com base em tentativa-erro, isto é, se o KDC principal não responder, tentar no KDC alternativo e assim sucessivamente até obter sucesso ou falha após percorrer todos os KDC
    - Esta funcionalidade está contudo disponível no Active Directory da Microsoft
- Caso o KDC principal esteja inoperacional, a administração do Kerberos não pode ser efetuada até que ele seja restaurado

Administração de Sistemas ASIST

Tópico 4

Criptografia

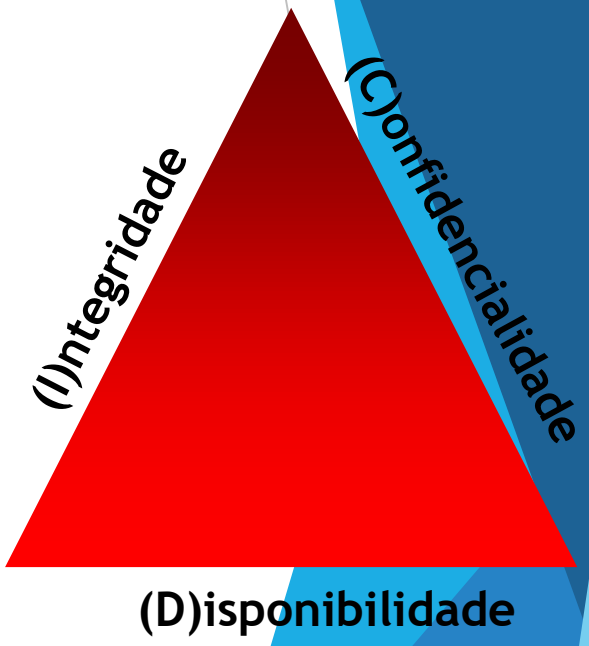
Pinto Leite, Jorge (jpl@isep.ipp.pt)

Criptografia

- O bem mais valioso que possuímos são os dados
- Por maior que seja o custo, um sistema e/ou uma infraestrutura pode ser substituída, mas tal não se aplica necessariamente aos dados
- O custo de perda dos dados, ou a perda de confiança neles, não é quantificável e muitas vezes impossível, o que acarreta inúmeros efeitos secundários
  - Legais, fiscais, operacionais, de reputação, …
- E, por vezes, são secretos ou privados, pelo que o acesso por terceiros acarreta perda de credibilidade e fugas de informação
- Ou seja, os dados são essenciais à operação de cada um, organização ou pessoal, e importa que sejam
  - **Credíveis**
  - **Disponíveis**
  - **Confidenciais**

Criptografia

- Note-se que estes princípios não se aplicam da mesma forma a todas as áreas da infraestrutura (rede, equipamentos, dados)
- Dentro dos princípios anunciados, a *disponibilidade* deve ser interpretada como *estar disponível na altura (e/ou espaço de tempo) apropriado*
- Entende-se por segurança os conceitos necessários para que a informação esteja disponível, certa e legível apenas a quem está autorizado
- Este conceito pode ser (e é) estendido à própria infraestrutura com as devidas alterações ou adaptações aplicáveis
- A tríade CID (ou CIA em inglês) espelha a base da segurança



Criptografia

- A confidencialidade
  - Garante que existe o nível apropriado de secretismo em cada nó de processamento e que se previnem as fugas de informação
  - Deve existir em todo o sistema e não apenas em algumas partes
  - Pode ser obtida através de
    - Encriptação dos dados guardados e transmitidos
    - Comunicações seguras
  - Pode ser anulada através de
    - Monitorização das comunicações
    - Engenharia social
    - Roubo de credenciais de autenticação
- Sem confidencialidade, os dados podem ser lidos por quem não tem a autorização apropriada

Criptografia

- A integridade
  - Garante a exatidão e fiabilidade do sistema e na prevenção da alteração não autorizada de dados
  - Garante que ataques e erros não comprometem a informação e o sistema
  - Pode ser obtida através de:
    - Boa gestão das capacidades dos agentes do sistema
    - Mecanismos de deteção de intrusão
    - Controlos de acesso apropriados
- Sem garantia de integridade, um sistema pode funcionar sobre dados incorretos sem o saber

Criptografia

- A disponibilidade
  - Garante a capacidade necessária a fim de atuar de forma previsível e com desempenho aceitável
- Ações relevantes que podem ser tomadas para a garantir são
  - Evitar pontos únicos de falha
  - Tomar medidas de salvaguarda da informação
  - Incluir mecanismos de redundância
  - Garantir a disponibilidade a quem dela precisa
- Pode ser afetada por ataques de negação de serviço

Criptografia

- Os mecanismos para garantir os dois primeiros princípios (confidencialidade e integridade) baseiam-se na utilização da criptografia
- O terceiro princípio (disponibilidade) tem como premissa mecanismos diferentes
  - Afinal o que se quer garantir com este princípio é que um utilizador autorizado obtém acesso aos dados existentes no sistema num tempo menor ou igual ao SLA definido
- Não se pretende aqui um curso completo de criptografia mas apenas os seus princípios básicos

Criptografia

- Criptografia – kryptós (escondido) + gráphein (escrita)
- Ciência e arte de ocultar texto (informação) de forma a que se torne indecifrável por terceiros - objetivo quando começou a ser aplicada
- Utilizada desde há milhares de anos (etruscos, romanos, etc.)
- Tendo um texto (**P**) e aplicando a cifra com uma chave(**K**) obtém-se o texto cifrado (**C**)
  - $\mathbf{C = P \oplus K^{+}}$ ou $\mathbf{C = E(P,K)}$ ou $\mathbf{C = Ke(P)}$
  - $\mathbf{P = C \oplus K^{-}}$ ou $\mathbf{P = D(C,K)}$ ou $\mathbf{P = Kd(C)}$
- Utilizada atualmente quer para tornar indecifrável o texto para entidades não autorizadas quer para o autenticar

Criptografia

- Um sistema criptográfico ideal deve ser difícil de quebrar, fácil de utilizar, robusto (ou seja, dois textos diferentes não devem resultar na mesma cifra se a chave utilizada é a mesma) e fácil de substituir (se quebrado)
- Note-se o *difícil de quebrar*
- A definição mais exata é assumir *difícil* como *impraticável*

Criptografia

- A criptografia por si só é bastante inútil
  - Tem de fazer parte de um sistema mais vasto
- Compare-se a criptografia com uma fechadura
  - Por si só, é inútil
  - Mas se fechar um acesso a algo exceto para quem possuir a chave, torna-se importante
  - Se assumirmos que o *algo* a que dá acesso se estende a pessoas, o problema assume outras dimensões comportamentais
    - Essas pessoas precisam lembrar-se de fechar a fechadura quando saem e
    - Precisam lembrar-se de não deixar a chave à vista ou disponível

Criptografia

- E tal como nas fechaduras, há a *boa* criptografia e a *má* criptografia
- Além disso, a criptografia tem que distinguir entre *bom* e *mau* acesso
  - O que é muito mais difícil do que manter todos de um lado apenas
- Um aspeto a manter em mente é que não é razoável aplicar criptografia "boa" ao mesmo tempo que se deixam pontos fracos na infraestrutura
  - Imaginem um cofre de um banco com uma porta e uma fechadura fantásticas mas que tem uma janela dentro do cofre, com ou sem grades
- Assumamos então um princípio base

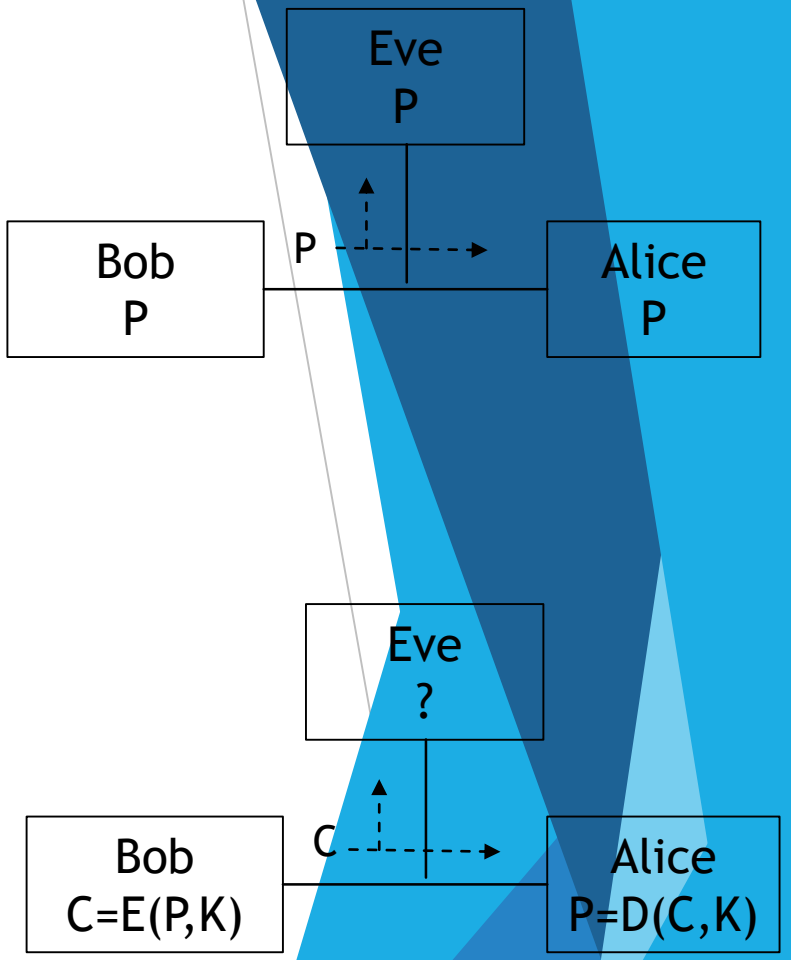
  A segurança de um sistema é tão forte quanto o seu ponto mais fraco
- E um outro ainda

  A criptografia não é a solução para os problemas de segurança

  (mas é parte da solução)

Criptografia

- O primeiro objetivo da criptografia foi encriptar/cifrar as mensagens
- Bob e Alice querem trocar mensagens (**P**) entre eles mas sem que Eve, que está à escuta, as entenda
- O que podem fazer é combinar um segredo entre eles (segredo partilhado) que será a chave (**K**) e encriptar cada mensagem enviada aplicando um algoritmo com essa chave, obtendo assim o texto cifrado **C**
- O destinatário da mensagem decifra a mensagem aplicando a mesma chave **K** mas a função inversa
- Cada interveniente na mensagem necessita saber o algoritmo a aplicar e a chave
- Este exemplo aplica-se a dados em comunicação mas pode e deve ser aplicado também a dados armazenados no sistema



Criptografia

- É claro que há custos e constrangimentos no processo
  - Que algoritmo de encriptação/desencriptação escolher?
  - Quão robusta é a chave (segredo)?
- Kerckhoffs definiu em 1883 vários princípios aplicáveis à criptografia que ainda se aplicam – embora adaptados aos dias de hoje
- Um dos princípios é ***A segurança de um sistema de encriptação deve depender apenas do segredo da chave e não do segredo do algoritmo***
- Há várias razões para isto
  - Os algoritmos são difíceis de desenvolver e alterar
  - Algoritmos secretos são difíceis de distribuir (embora com algumas exceções)
- Atualmente, não basta o secretismo da mensagem
  - Torna-se necessário garantir a integridade e a autenticidade
  - E também, o não-repúdio (basta pensar nos bancos e na autoridade tributária)

Criptografia

- Além das considerações anteriores, uma outra emerge
  - Qual o procedimento a adotar se o algoritmo for quebrado?
- Vários autores escreveram sobre o tema
  - ***The enemy knows the system*** (Claude Shannon)
  - ***There are two ways to design a system. One is to make it so simple there are obviously no deficiencies. The other is to make it so complex there are no obvious deficiencies.*** (C. A. R. Hoare, citado em Kaufman et al., Network security, p. 441)
  - ***Criptography is rarely ever the solution to a security problem.*** (D. Gollmann,Computer Security, p. 203)

Criptografia

- Os sistemas criptográficos podem ser tipificados por
  - **Algoritmo**
    - Algoritmo secreto (usado na área militar)
    - Algoritmo conhecido
  - **Segredo/Chave**
    - Simétrica (segredo partilhado)
    - Assimétrica (chaves de encriptação e desencriptação diferentes)
  - **Método**
    - Bloco
    - Contínuo (*stream*)
  - **Aplicação**
    - Reversível/bidirecional (Integridade, Confidencialidade, Autenticidade)
    - Irreversível/unidirecional (Integridade, Autenticidade)

Criptografia

- Já foram descritos alguns constrangimentos na utilização de um algoritmo secreto
- De realçar que o desenvolvimento de um algoritmo secreto, embora possível, tem inconvenientes
  - Quão seguro é?
  - Como se distribui?
  - Que garantias temos de robustez, isto é, que dois textos diferentes não produzam a mesma cifra?
  - Quem faz a sua criptoanálise?
- Se adotarmos um algoritmo conhecido temos a garantia que já foi feita a sua criptoanálise e, provavelmente e caso aplicável, das suas implementações

Criptografia

- Pensando no segredo, a forma mais simples e usual é o segredo partilhado
  - É usado, como visto em tópicos anteriores, nas senhas de acesso
- Um método possível é a **substituição**, que consiste em substituir cada caractere do texto por um diferente
  - Este é o método utilizado na Cifra de César
  - Esta cifra consistia em substituir cada caractere por um outro 3 (três) posiçõesno alfabeto para a direita (não era exatamente só isso, mas assumamos dessa forma)
- Se o texto cifrado for **C = ehpylqgrvdqryddxodghdvlvw**, qual será a mensagem **P**?
  - **$K^+ = 3 \leftrightarrow K^- = -3$**
  - Cifrar: **a → d; e → h; z → c; …**
  - Decifrar: **e ← b; h ← e; c ← z; …**

Criptografia

- Em alternativa à substituição, pode ser feita a **<u>transposição</u>** (ou permutação), que se baseia na alteração da ordem dos caracteres do texto
  - Texto: **Hoje temos aula de ASIST**
  - Texto alterado: **H J t m s u a e S S**

    o e e o a l d A I T
  - Cifra: **HJtmsuaeSSoeeoaldAIT**

  - Chave: **4132** (vamos acrescentar um "**k**" ao texto para que o seu tamanho seja múltiplo do tamanho da chave)
  - Cifra: **eHjootmelasuaSdAekITS**

Criptografia

- O segredo partilhado é apetecível por vários motivos, dos quais se destaca ser o método com melhor desempenho nas operações de cifragem e decifragem
- Contudo, tem vários inconvenientes
  - Para um segredo pré-partilhado ser efetivo, deve ser exclusivo de um par de entidades
    - Ou seja, para N entidades deverão existir $\mathbf{C}_{(n,2)}$ (combinações de N dois a dois)
  - O elevado número de segredos conduz a problemas humanos
    - Para 100 (cem) entidades, são necessários $\mathbf{C}_{(100,2)} = \frac{\mathbf{100!}}{\mathbf{2!*(100-2)!}} = \mathbf{4950}$ segredos!
  - Também num contexto prático, como partilhamos o segredo com uma entidade? Envio por SMS?

Criptografia

- A Cifra de César tem o inconveniente que caracteres iguais produzem cifras iguais
- A Cifra de Vigenère baseia-se no princípio da Cifra de César mas alterando os caracteres de forma não constante
- Em vez disso, utiliza uma matriz de caracteres e uma chave, substituindo cada letra do texto P pelo caractere na intersecção do caractere do texto com o caractere correspondente da chave
- Exemplo:
  - Texto: **Qual o dia da semana? → qualodiadasemana?**
  - Chave: **asist → asistasistasista**
  - Cifra: **qmidhdaivtswusga?**

2 2 1 1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Cifrar: caractere na intersecção da linha da letra correspondente da chave com a coluna da letra do texto

Decifrar: letra da coluna que contém na intersecção da linha da letra correspondente da chave o caractere da cifra

Criptografia

- Em alternativa há a cifra assimétrica
- Baseia-se na utilização de algoritmos e chaves diferentes para a encriptação e desencriptação
- Na cifra assimétrica **Ke <> Kd**
- Também é conhecida por **cifra de chave pública** porque **Ke** pode ser tornado público já que apenas a **Kd** (secreta) permite desencriptar
  - Na verdade, um texto claro pode ser encriptado com a chave **Kd** e desencriptado com a chave **Ke**
- Este método é pesado em termos de processamento, portanto inadequado para grande volume de dados
- Ao mesmo tempo, é fácil de utilizar pois não exige a troca de chave (segredo)
- Utilizado frequentemente para negociar de forma segura um segredo partilhado

Criptografia

- A minha chave pública:

-----BEGIN PGP PUBLIC KEY BLOCK-----

mQGNBGJQU0EBDACrwZQ98o5Lxk18HefD6bC7ri1f7QgmgoyiaDHa5SmsnCxL72LI
WQuPzEaXD4oQe+YqNFQ9HSspE/PaZw9gouYEXnOdZygtEpbbTXqUR+SmecQr637I
kGBWRXAX5+1NR7Od0S8jNoT96XzDyO80ZMlaIeaXVRLnhJRkIUdA5RnFCt85/jWm
+7jAMb0woPVzMeHTSGPhrEfYO+qLAjPq6yyqjDCkj1RjltoEKGaaFKesJt6p6deM
sCkSMOpOiWmBI71gVuSzgPnt63JXaL1ZubnkHyeglL4fj/j9ewbvW+Gwc6Ic9MDL
jkwp/3NH1NwelIcHi1stpg/FMwrWQqN6w+SNCbPixG9hF7oR7rvSiw9UDEqYY9SJ
OdjR6U1DIkZKs2IwyqEEHXXCrEPeaAYujZTEDufYfrXiBh0ocHqEdP2d4KK4psQ8
x/JbjmzxFX0MVUCiPSGk4wc9+90yXkXOVEBC9udKMkareFyCyBqjPH6SqqExIrc/
3DzRNoTv6vTsnssAEQEAAbQjSm9yZ2UgUGludG8gTGVpdGUgPGpwbEBpc2VwLmlw
cC5wdD6JAc4EEwEIADgCGwMFCwkIBwIGFQoJCAsCBBYCAwECHgECF4AWIQQ73in2
iCYMAh6y9CNwcBn0rY848wUCYlBTrgAKCRBwcBn0rY848z7aC/sHQXIpIQRM8Vfr
ZQupxZvlPD7i0KJdmnMkJckB6pJmF4IXn7dvOoaseRDFRWtWVWj6YtY/QOkncpM2
s/Ij7PWRda20a+qPqLYhxog5zMokkX2duac1aQqIDBiXuiC2yl9cELTxH9IRDhhj
uxJVZE7sb6Bennll2U/rV2wuWSiUlx5cw2x0t/2EKtVlnVjn79uyzj+8n5/C68kx
cRtzHrvxCv/8RqfcwmO3x01cOm8HkTvrD1O/93nxfys9Gi6DqjwgZHDZwXfx4jFB
Mw7Q4NH1rACkShH5F+7Z5h68bpWu2c3DdrFazqUOO37EE4RA7Cb60y13feSIJ6JS
NvNN4vlL1so5pBMrBu1b3NqkPVKm7Yn62h6CfdlCQrO+RNtNDTgZuj2q897Yjmx3
OdZNQasX8XjO9b5EDALIChdvC6h10bKozimHFzjSO85fRN19/Mi12KsB7FwInmiH
0dpsl4EUD5gOLQRIXpuphUVgQDCh68Xq+J/EVZMlXtrz5enEKz25AY0EYlBTQQEM
AMpdZr8OoDFvNvUIQ0rnpaH1pFud6nLkPZveDshmmHcrQlBj9MNm9/yhKjHyK1+S
xcAVbfkRw+aKk0KGW4MjRa9+ky/DQC7U1ZGBiTv+LihtZWxb6qDS2dWXa7bnBD8y
i5DwB6ZQocOhtCoUu+UkHv57O8Etp2/dQ24P8nxCyMbfF4OvCEH6jO3tlE4Rds7o
NRmNyIEdP8/mCSJC69Joi63ZdvWO6ySQBW6UEpzS5ohCyzvujhO2FAZaHOX7bz8M
MauyoR89EWGUW0dgwKWMqInY89XbyYs+LI1qHv2XRfjlPT0ksEOillLxMfcCC7jv
nz3ppxKKltRFp0/vhm7g++0u9XJfJutzHN7cHcnRApDHqOknMspLuHM8eoSF9w9i
NLdvKu1F9x/7jLZ5CD5JMISHTVVFakrVdtzSKrFCe2x1bgOYirgTO5TENSqcFPWc
KEqxV6RAu1dQ+sD72uNLQs2YnzLXFPz5EHqs0gp774gQN7OxPPrPgRl6BTYphGRt
CQARAQABiQG8BBgBCAAmFiEEO94p9ogmDAIesvQjcHAZ9K2POPMFAmJQU0ECGwwF
CQPDeu8ACgkQcHAZ9K2POPOThQv/XqUCc2MqpVldbWC2pAl0HgL1/2QfMcSnsNtt
HMn0a83+bUKJ+N1C3vZd+rjegTAFYwC/noK+L6viOqIqwbZ50kk15hSEKffDxjG/
N6rdIMMn9wO3vJGisvd11fD0oPCLfZwbSGmKsIruGV+fgJTWL6HlGBN3FBFS4qT+
4jC7Y+aNdT1aUU2CKcvKJ9K4AFtBaoro4hQ1Ej7zJpQ4oUzawXBcoDaUIcQMIRG7
MntAV3YHB9s+XhVUQuHZLSSi6C2B2wnkms6WH2+IiO/Y1ztm8p92iP8P7YyBJFbJ
r7BUc5OUcfEqu4t3IDCDvTS6ch7PNgumGcgSY+sTCnnO2ZXt3oV5bIF6ga7pgRZn
Bc3gG7Rnn+qZ1stIdXrBd2s+9wAYtBRIXbbx044wZf9A7DSz//yJ62BXZ1lPiRaC
56U1OSRA1VvINaHIV1UFsDLK8fEHwZeuy+AfD+A6PS62ymy9sfI7VKzZCeCNTmWI
C8qYlENregzy03iQJnx22D8emn/z
=sSOT
-----END PGP PUBLIC KEY BLOCK-----

Criptografia

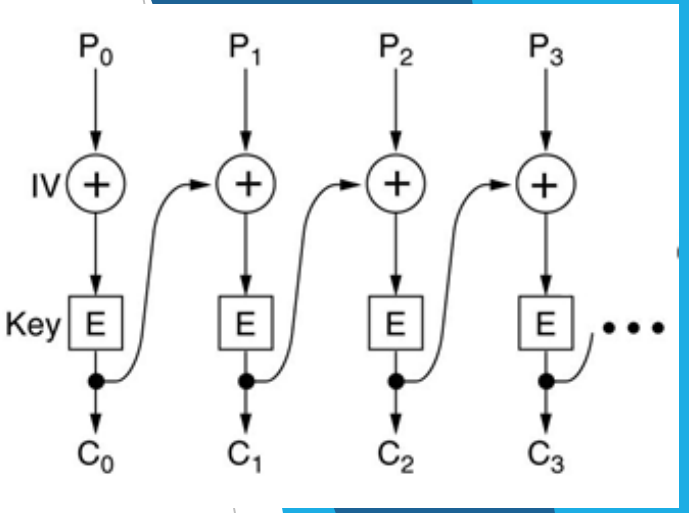
- Já referimos que o método de cifra pode ser bloco ou contínua (*stream*)
- No caso do bloco, o texto original é dividido em partes de igual tamanho, ou seja, $\mathbf{P = P_1P_2P_3...P_N}$
- Cada uma das partes é cifrada com a mesma chave **K**
- É adicionado um *padding* ao texto ou truncado no bloco final a chave para que as dimensões das partes (texto e chaves repetidas) seja igual
- Na cifra em bloco $\mathbf{C = K(P_1)K(P_2)K(P_3)...K(P_N)}$

Criptografia

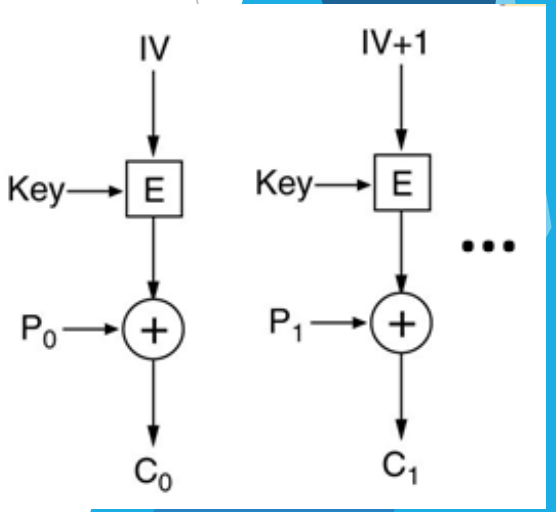
- Exemplo de aplicação da cifra em bloco
  - Texto P: **Vamos experimentar uma cifra**
  - Chave: **fcp** (método de Vigenère)
  - Texto P dividido em blocos: **vam ose xpe rim ent aru mac ifr a**
  - <u>Hipótese 1</u>: acrescentar um *padding* (seja **te**)
    - Texto P + *padding* : **vam ose xpe rim ent aru mac ifr ate**
    - Chave: **fcp fcp fcp fcp fcp fcp fcp fcp fcp**
    - Cifra: **acb tut crt wkb jpi ftj rcr nhg fvt**
  - <u>Hipótese 2</u>: truncar a chave no bloco final
    - Texto P: **vam ose xpe rim ent aru mac ifr a**
    - Chave: **fcp fcp fcp fcp fcp fcp fcp fcp f**
    - Cifra: **acb tut crt wkb jpi ftj rcr nhg f**
  - Como era de esperar todos os blocos de cifra são iguais variando apenas o último

Criptografia

- Este método tem como inconveniente que texto igual implica cifra igual
  - O que torna possível por exemplo ataques que procurem a frequência de cada caractere no idioma aplicado
- Designa-se por ***Electronic Code Book*** (**ECB**) este método
- Em alternativa podem-se utilizar métodos como
  - ***Cipher Block Chaining*** (**CBC**)
    - A cifra de cada bloco (excluindo o primeiro) é aplicada ao bloco seguinte (usualmente através de uma operação **XOR**) e o resultado é cifrado
  - ***Counter*** (**CTR**)
    - É aplicado um vetor de inicialização à chave e o resultado dessa aplicação é que é usada como chave para o bloco
    - O vetor inicial até pode ser zero, mas os seguintes são incrementados de um valor pré-determinado
  - Há outros métodos mas o objetivo é similar



Encriptação CBC



Encriptação CTR

Criptografia

- No caso **contínuo**, o texto original é dividido em partes de igual tamanho, ou seja, $P = P_1P_2P_3...P_N$ em que $\forall x \in (2, N) sizeof(P_x) = sizeof(P_{x-1})$
- Cada uma das partes é cifrada com uma chave diferente $K = K_1K_2K_3...K_N$
- O número de chaves **K** pode ser menor que o número de partes do texto **P**, caso em que uma chave pode ser repetida
  - Assumindo-se dessa forma como **chave periódica**
- Na cifra contínua $C = K_1 (P_1)K_2 (P_2)K_3 (P_3)...K_N (P_N)$
- E, se periódica, pode ser por exemplo $C = K_1 (P_1)K_2 (P_2)K_1 (P_3)...K_1 (P_N)$

Criptografia

- O problema aqui é sempre a chave, a sua dimensão, a maior ou menor dificuldade de um terceiro em obter o texto **P** tendo acesso a **C**
- Define-se o espaço da chave (***keyspace***) como o número de chaves possíveis dada a sua dimensão e caracteres permitidos em cada posição
  - Código de um cartão bancário
    - Chave de 4 caracteres numéricos
    - Cada posição pode assumir 10 valores distintos (zero a nove)
    - Keyspace = 10^4 = 10000
- Assumindo uma mensagem em português em que a chave usa todos os caracteres maiúsculos (A a Z incluindo K, W e Y) e tem a mesma dimensão (**N**) que o texto a cifrar, o espaço é $\mathbf{N^{26}}$, o que torna impraticável decifrar exceto a quem possuir a chave
- Este é o princípio do ***One Time Pad*** (**OTP**)

Criptografia

- Quanto à aplicação, a criptografia pode ser
  - **Reversível/bidirecional** também chamada ***two-way***
  - **Irreversível/unidirecional** também chamada ***one-way***

- Considera-se uma aplicação reversível/bidirecional se é possível obter o texto **P** tendo acesso à cifra **C**

- Considera-se uma aplicação irreversível/unidirecional se é impossível obter o texto **P** tendo acesso à cifra **C**
  - Ou, melhor dizendo, impraticável

Criptografia

- Se a cifra é bidirecional deve ser
  - **Simples**
    - A codificação e descodificação de **P** é fácil dada a chave **K**
  - **Resistente**
    - Dado um texto **P** e o correspondente **C** é impraticável calcular **K**
  - **Unicidade (resistência à colisão)**
    - Dado um texto **P** e o correspondente **C** é impraticável calcular uma chave **K'** tal que **Ek(P) = Ek'(P)**

Criptografia

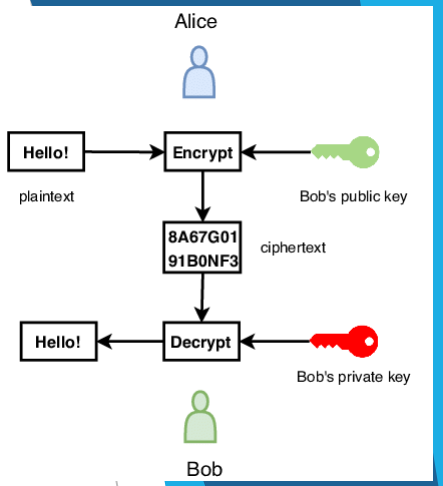
- Se a cifra é <u>unidirecional</u> deve ser
  - **Simples**
    - A codificação de **P** é fácil dada a chave **K** e provoca um resultado com comprimento fixo **H**
  - **Irreversível**
    - É impraticável inverter **H** ($P \neq H^{-1}(C)$)
  - **Unicidade (resistência à colisão)**
    - Dado um texto **P1** qualquer é impraticável encontrar outro **P2** tal que **H(P1) = H(P2)**
    - Se a resistência à colisão for fraca, então pode existir um **P1** e um **P2** tais que **H(P1) = H(P2)**

Criptografia

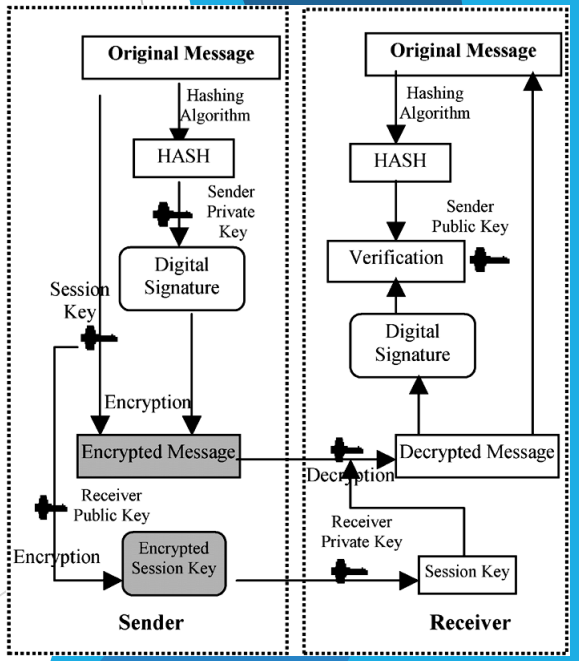
- Mas porquê a cifra unidirecional?
- Como referido, inicialmente o objetivo da criptografia era a ocultação ou obscuração de um texto através da cifragem
- Mas se essa operação torna impraticável a compreensão da cifra por um terceiro não autorizado, não impede que esse terceiro altere alguns caracteres da cifra – que até podem resultar ao decifrar num texto válido (embora necessariamente díspar do texto pretendido)
- A finalidade da cifra unidirecional é autenticar o texto gerando um ***hash*** ou ***Message Authentication Code*** (**MAC**) que é apenso à cifra **C**
  - O MAC tem um tamanho fixo que depende do algoritmo criptográfico usado

Criptografia

- A utilização de cifra assimétrica possibilitou o **não-repúdio**, propriedade já referida (slide 13)
- Se o *hash* gerado for assinado com a chave privada do emissor
  - O recetor pode desencriptá-la com a chave pública do emissor e verificar a integridade da mensagem
  - O emissor não pode repudiar a validade ou autoria da mensagem
  - Designa-se por **assinatura digital** o *hash* assinado com a chave privada do emissor
- O não-repúdio pode ser conseguido por outras formas mas com custos de processamento
  - Cifra com a chave pública do recetor, depois com a chave privada do emissor
  - Decifra com a chave pública do emissor, depois com a chave privada do recetor



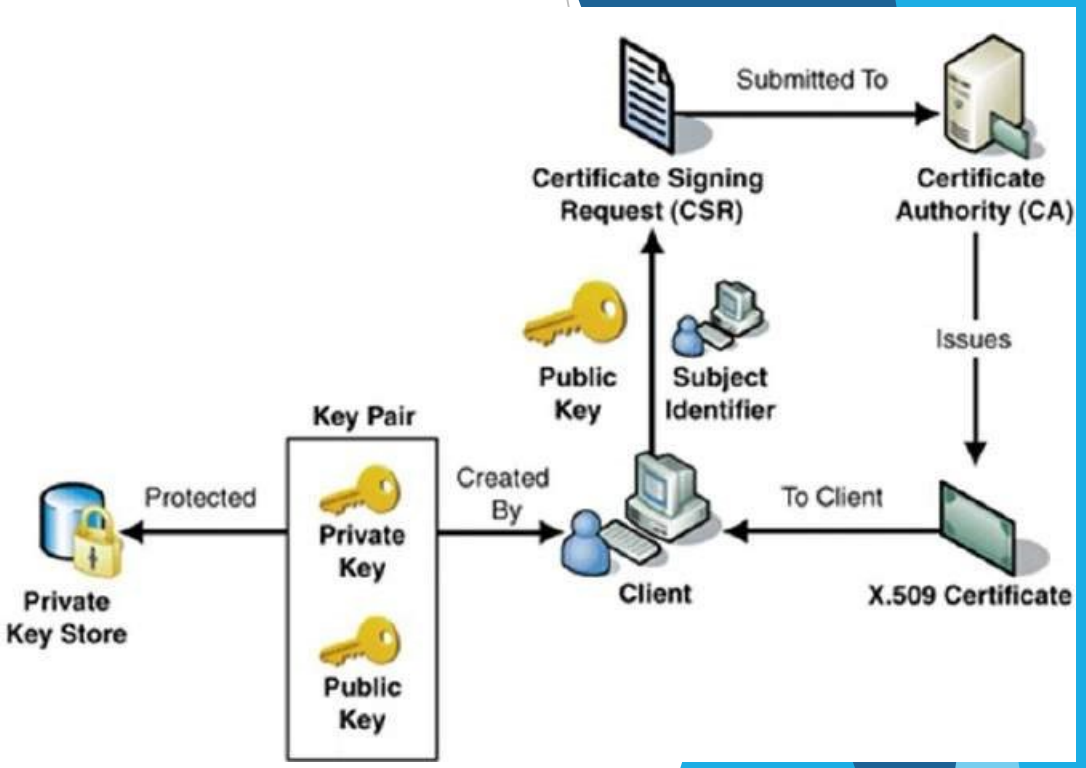
Fonte: ResearchGate



Fonte: ResearchGate

Criptografia

- Como exemplo da cifra assimétrica veja-se a ***Public Key Infrastructure*** (**PKI**)
- Possibilitou o uso da criptografia sem a problemática da troca de chaves
- A **PKI** consiste num conjunto de regras, procedimentos, políticas, aplicações e instalações física e logicamente seguras
- O objetivo é torná-la confiável para gerar, gerir, distribuir, guardar, usar e revogar certificados digitais e as chaves digitais
- Este tema será aprofundado nas aulas teórico-práticas



Fonte: TutorialsPoint

Criptografia

- Já foi referido que a cifra assimétrica não é apropriada para grandes volumes de dados
- Contudo tem vantagens significativas na distribuição de chaves, na autenticidade do emissor/recetor, etc.
- Porque não usar uma mistura de ambas?
  - Recorrendo à cifra assimétrica negociar um segredo partilhado entre as partes
    - Ou seja, a possibilidade de usar uma cifra simétrica
  - Usar o segredo partilhado negociado para cifrar e decifrar a informação

Criptografia

- A aplicação da tríade de segurança (Confidencialidade, Integridade, Disponibilidade) não se esgota na comunicação entre partes
- A informação armazenada nos dispositivos de armazenamento também necessitam ser sujeitos a operações criptográficas para
  - Auxiliar a confidencialidade dos dados
  - Validar a integridade dos dados
- No que toca às comunicações, estes algoritmos são utilizados em aplicações e/ou protocolos que substituíram versões mais antigas e atualmente consideradas inseguras
  - Alguns exemplos são **SSH**, **VPN**, **TLS**, **IPsec** e a norma **802.11i**
- De notar que a aplicação de um algoritmo criptográfico numa camada não inviabiliza a aplicação de um outro numa camada diferente, embora tenha um custo no processamento

Criptografia

- A aplicação da criptografia deve ser planeada e implementada de forma correta
- Há normalmente uma tendência para imaginar e considerar a criptografia apenas durante a comunicação de dados
- Não há dúvida que nessas circunstâncias deve ser aplicada mas se e só se a informação que circula é sensível
  - Não há necessidade de cifrar toda a informação até porque isso representa um custo do ponto de vista de ocupação de largura de banda e desempenho da infraestrutura
- Mas não esquecer que os dados armazenados também podem e devem ser protegidos recorrendo à criptografia
  - Aplicando o mesmo princípio de proporcionalidade do que deve ou não ser protegido

Administração de Sistemas ASIST

Tópico 5

Segurança e serviços

Pinto Leite, Jorge (jpl@isep.ipp.pt)

Segurança e serviços

- As redes são um meio que é necessário proteger de ataques
- São por vezes muito extensas, tornando complicado assegurar a inexistência de acessos físicos indevidos
- Estão normalmente ligadas a acessos externos, seja para acesso à Internet seja para providenciar o acesso remoto dos utilizadores autorizados, pelo que nesses casos há sempre um ponto comum à rede interna (intranet) e a uma rede externa (Internet)
- Esse ponto comum delimita o perímetro da rede interna e onde se situa, habitualmente, a sua defesa
- Designa-se neste caso por **defesa de perímetro**
- A situação atual, com o aumento das redes sem fios, a necessidade de proteger a infraestrutura de ataques por engenharia social aos utilizadores válidos ou até por forma a mitigar ações maldosas internas propositadas, levou à necessidade de alargar o *perímetro* e à adoção da **defesa em profundidade**

Segurança e serviços

- É importante alterar ideias pré-estabelecidas e evoluir da ***defesa de perímetro*** para uma ***defesa em profundidade***

- A defesa em profundidade é um conceito que aplica várias camadas de proteção
  - A ideia é que se uma delas falhar continuam a existir outras que asseguram a defesa
- Com este princípio acordemos que cifrar e autenticar a informação sensível armazenada na infraestrutura é uma ideia a considerar
- Claro que tem custo de processamento… que de acordo com o BCM deve ser ponderado face ao benefício obtido

Segurança e serviços

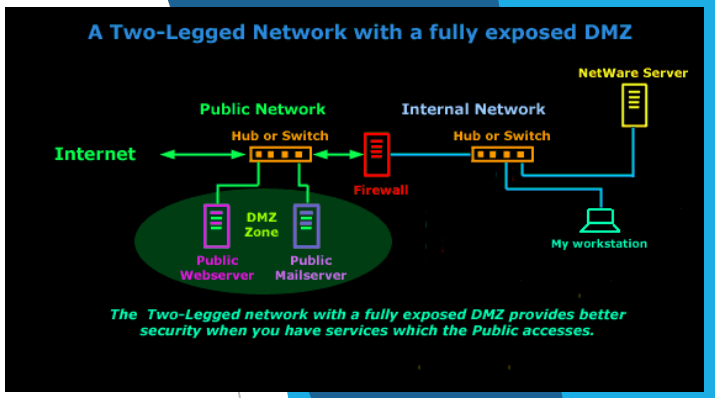
- Uma infraestrutura de rede é habitualmente dividida em três zonas:
  - A rede do CPD, onde se situam os servidores de acesso externo e a base da infraestrutura de rede
  - A rede interna, onde se situam os servidores exclusivamente internos, os postos de trabalho e ativos que garantem o funcionamento e interligação destes quer ao servidores internos quer ao exterior, se aplicável
  - As redes externas e os dispositivos que asseguram o acesso a elas
- A delimitar estas redes ou pelo menos algumas destas redes estão ativos que evitam ou mitigam ações indevidas, designados por *firewalls*
- Os servidores de acesso público podem ser os mais problemáticos
  - Por um lado necessitam assegurar o acesso público aos serviços pretendidos
  - Por outro lado podem potenciar pontos de acesso a atacantes
- É por isso vulgar colocar servidores de acesso público numa rede semi isolada da rede interna que se designa por ***Zona Desmilitarizada*** (**DMZ** do termo em inglês *Demilitarized Zone*)

Segurança e serviços

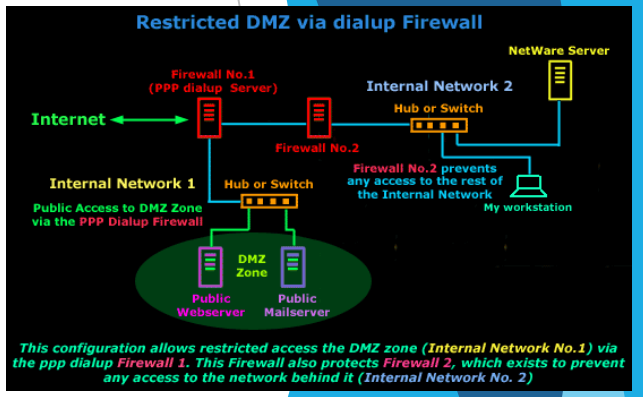
- O recurso à segmentação de redes por VLAN não oferece garantias de segurança por essa segmentação ser lógica
  - Por isso e independentemente de se proceder a esta segmentação, deve ser sempre considerado como um meio de transmissão partilhado
- As *firewalls* são habitualmente instaladas em *routers* por serem os pontos de fronteira entre as redes quer internas (Intranet e/ou CPD) quer externas (Internet)
  - Até por os *routers* terem a possibilidade de definir ACL's e serem os dispositivos que encaminham os pacotes através da tabela de *routing*
- Existem *firewalls* dedicadas *(appliance)* mas é também uma aplicação que pode ser instalada ou configurada num *router* ou até num posto

Segurança e serviços

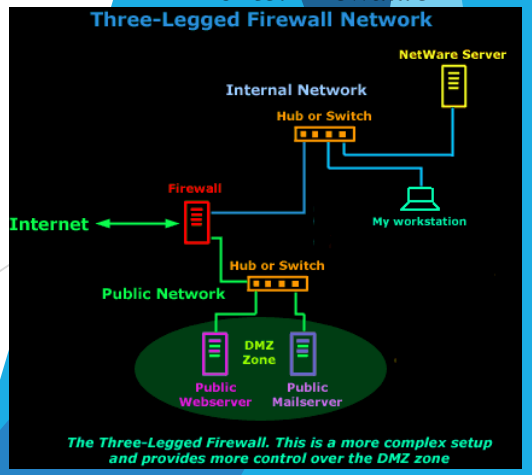
- A segmentação de uma rede sensível deve ser sempre protegida por uma *firewall*
- Há várias formas de implementar a *firewall* de acordo com a necessidade e objetivo
- A topologia representa o número de redes a que se liga
- Na situação apresentada na figura de cima *(twolegged)*, a rede interna está protegida pela *firewall* mas com uma DMZ totalmente exposta e desprotegida
- Na figura do meio *(*duas firewall *twolegged)*, a DMZ passa a estar protegida com recurso a duas *firewall*
- Mas na figura de baixo *(threelegged)* a mesma funcionalidade é obtida embora com recurso a uma só *firewall*



Fonte: Firewall.cx



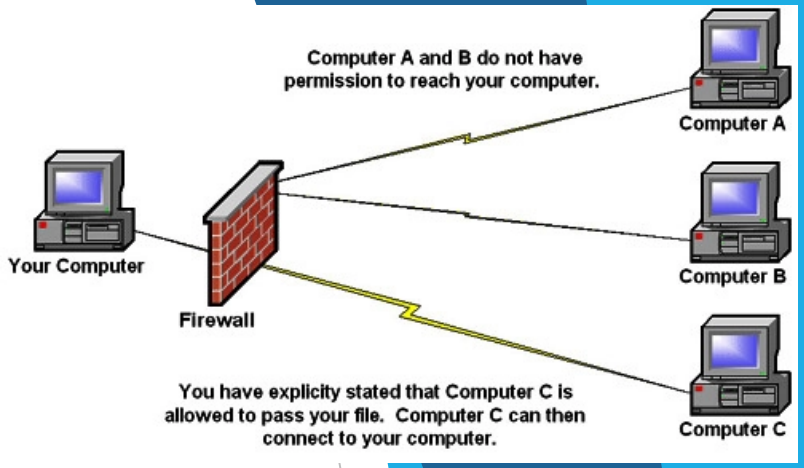
Fonte: Firewall.cx



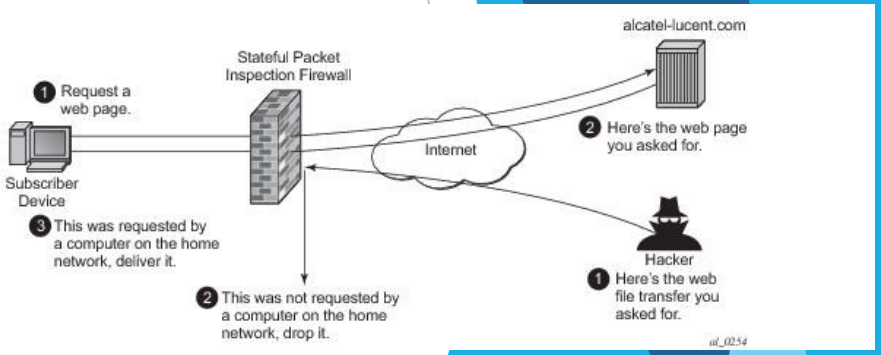
Fonte: Firewall.cx

Segurança e serviços

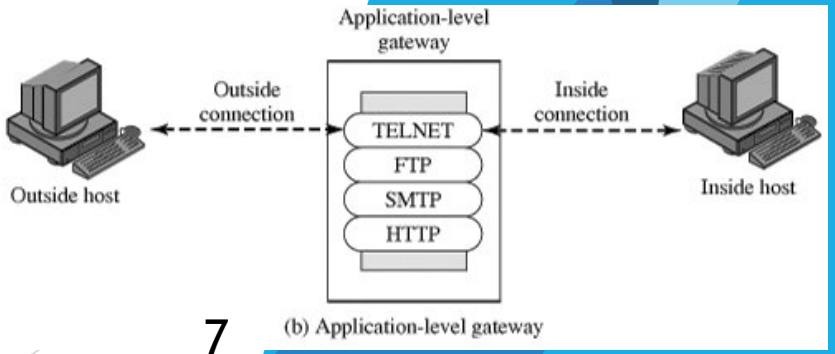
- O tipo representa a forma de atuação, como
- ***Packet-Filter*** ou ***firewall estática*** ou ***screening router***
  - Aplica um conjunto de regras estáticas a cada pacote IP que a atravessa
  - Baseia-se apenas no cabeçalho IP (nível 3 do modelo OSI)
- ***Stateful Packet Inspection***
  - Além das regras estáticas aplica regras dinâmicas temporárias, baseadas na configuração e no passado recente de comunicação
- ***Application-Level*** ou ***Proxy Server*** ou ***Bastion Host***
  - Retransmite os pacotes IP de acordo com o tipo de tráfego autorizado



Fonte: Quizlet



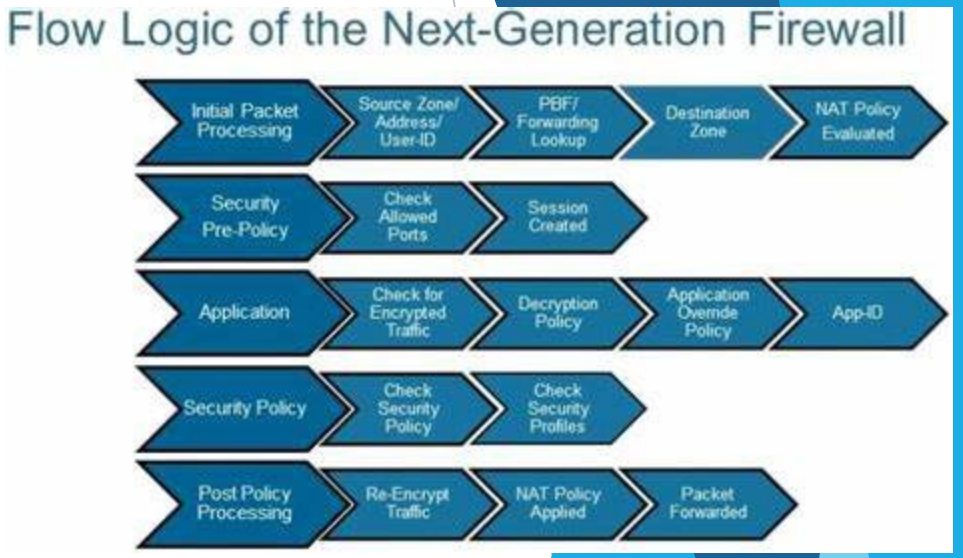
Fonte: Alcatel-Lucent



Fonte: Alcatel-Lucent

Segurança e serviços

- ***Next Generation Firewall*** **(NGFW)**
  - Mais recentes, incorporam as características da Packet Filter e da Stateful, e ainda controlo de aplicação, ***Intrusion Prevention System*** (**IPS**) e um ambiente (*sandbox*) onde conteúdo suspeito pode ser inspecionado



Fonte: Palo Alto

Segurança e serviços

- A *firewall* é um auxiliar útil na defesa de uma rede, contudo algumas considerações devem ser tidas em conta
  - Não protege ataques encapsulados em tráfego permitido
  - Não protege ataques internos pois só analisa e age perante o tráfego que a atravessa
  - Não protege contra vírus instalados em programas transferidos
  - Muitas vezes é configurada para permitir todo o tráfego proveniente da rede interna e age apenas no tráfego proveniente da rede externa
    - E nesses casos se não é permitido por ser uma resposta a um pedido anterior da rede interna
  - Ou seja, assume que todos os *bad guys* estão no exterior e todos os que estão no interior são *good guys*
- Se um utilizador interno for enganado e executar um código malicioso, esse código pode inibir a *firewall*

Segurança e serviços

- Se existirem erros de configuração ou dificuldade de reconhecer novas aplicações, pode tornar difícil ou mesmo impossível acessos legítimos
- O encapsulamento de um tipo de tráfego noutro (por exemplo, ftp encapsulado em http) pode colocar em causa as regras definidas pelo administrador de sistemas
- Finalmente, não esquecer que a *firewall* vai representar um peso no processamento

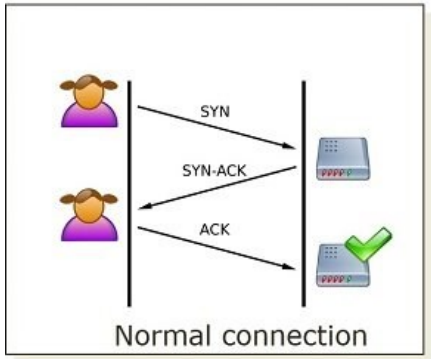
Segurança e serviços

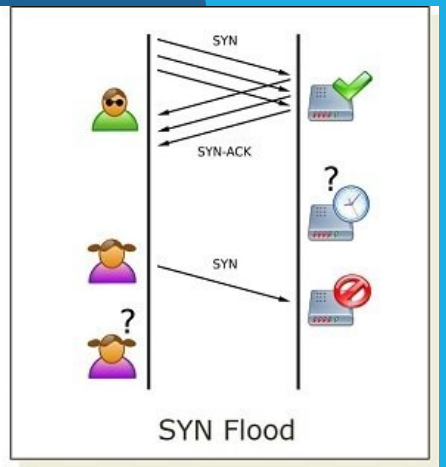
- Tomando em consideração as vantagens e fraquezas da *firewall*, a sua utilização é recomendada
- Mas o seu tipo e regras devem ser ajustados à zona que protege
  - Numa DMZ, **permitir apenas respostas a pedidos válidos** e **não iniciar pedidos** exceto em situações absolutamente necessárias e controladas
  - As permissões devem ser limitadas ao mínimo essencial a cada serviço
    - Não faz sentido (e poderia ser explorado por atacantes) permitir tráfego SMTP a um servidor que se dedica em exclusivo a HTTP por exemplo

Segurança e serviços

- Do lado interno da *firewall*, ou seja, na rede local, um utilizador pode, por maldade ou acaso, potenciar um conjunto de ações que violam a segurança da comunicação interna
- De entre as ações possíveis destacam-se a inspeção de tráfego (*sniffing*), a utilização de endereços IP inválidos (*spoofing*) e os ataques de interposição *(Man-in-the-Middle MITM)*
  - Através do *sniffing* inspeciona-se todo o tráfego do nível 2 até ao nível da comunicação em curso do modelo OSI
  - Através do *spoofing* podem-se desencadear ações como enganar outros sistemas (MAC spoofing, ARP spoofing, IP spoofing) levando-os a endereçar os pacotes enviados para sistemas que estão infetados
  - Recorrendo normalmente a *spoofing* previamente, um atacante pode levar os sistemas a recorrer a um servidor DNS inválido ou infetado

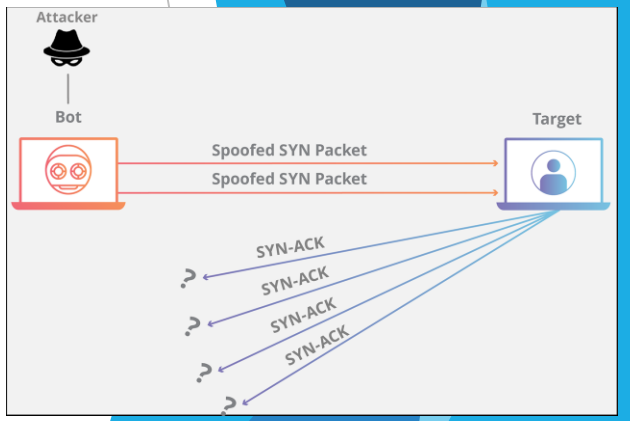
Segurança e serviços





Fonte: Firewall.cx

- Do lado externo, ter em conta os ataques ***Denial of Service*** (**DoS**) e ***Distributed Denial of Service*** (**DDoS**)
- Tratam-se de ataques que executam pedidos legítimos aos sistemas de acesso público, mas tirando partido da forma de funcionamento dos protocolos procuram esgotar os recursos dos sistemas internos
  - Uma *firewall* estática não consegue prevenir estes ataques
- A diferença entre um **DoS** e um **DDoS** é que o primeiro é proveniente de um único sistema (ainda que ferramentas permitam que ele simule outros, denominados *decoys* ou *zombies*) enquanto o segundo é uma ação concertada de múltiplos sistemas
  - É estimado que em 2022 decorram cerca de 14,5 milhões de ataques DDoS (fonte: Cybercrime magazine)
- Um ataque **DoS** ou **DDoS** a tirar partido do **SYN Flood** é apresentado nas imagens



Fonte: PureVPN

Segurança e serviços

- Do lado externo, permitir apenas e só o tráfego de acesso à rede interna que se pode iniciar externamente
  - Mas garantir que as respostas aos pedidos legítimos são asseguradas
- É responsabilidade do administrador de sistemas mitigar ou inibir o *spoofing*
  - Interno, que tenta sair da rede interna com um endereço de origem inválido nessa rede
  - Externo, que tenta entrar na rede interna com um endereço de origem da rede interna
    - Não é razoável que tráfego com origem externa mas com endereço IP da rede interna atravesse a *firewall*
    - Atacantes podem tentar esse método pois a regra é normalmente *quem é de dentro é de confiança*

Segurança e serviços

- Os ativos de rede são pontos fulcrais e cruciais do funcionamento da infraestrutura
- Por omissão, operam no modo *best-effort*, isto é, todo o tráfego é tratado da mesma forma e a ordem de saída é *first-in-first-out* (FIFO)
- Em situações de pressão isto implica que algum tráfego pode ser descartado – quando esgota a capacidade de armazenamento do ativo – ou atrasado – enquanto aguarda a sua vez de ser processado
- Contudo, pode haver tráfego considerado prioritário, sendo objetivo que seja processado de forma privilegiada
- Quais são então os atrasos que podem ocorrer na infraestrutura?

Segurança e serviços

- Os atrasos na infraestrutura (latência) e as variações estatísticas (*jitter*) desses atrasos, para além das perdas por esgotamento da capacidade dos ativos já descritas, conduzem à inexistência de uma garantia de continuidade de entrega do tráfego
- O atraso que um pacote sofre desde ser enviado até ser recebido no nó final é a soma de quatro atrasos[1]
  - **Atraso de processamento** onde ocorre a correção e processamento do cabeçalho
  - **Atraso de espera** nos ativos entre os nós emissor e recetor
  - **Atraso de transmissão** que depende da velocidade dos canais atravessados entre os nós emissor e recetor
  - **Atraso de propagação** que depende da distância entre os nós

[1] Adaptado de Prof. Legatheaux Martins

Segurança e serviços

- O atraso de processamento pode ser considerado constante
- O atraso de espera é função do tráfego total que o ativo tem de processar, sendo por isso aleatório; pode ser contudo estimado por

$$\boldsymbol{Atraso\ de\ espera} = \frac{\boldsymbol{Tam.\,médio\ pacotes\ (bits) x\ Taxa\ média\ de\ chegada}}{\boldsymbol{Velocidade\ de\ transmissão\ (bps)}}$$

- O atraso de transmissão é função da velocidade de cada segmento atravessado

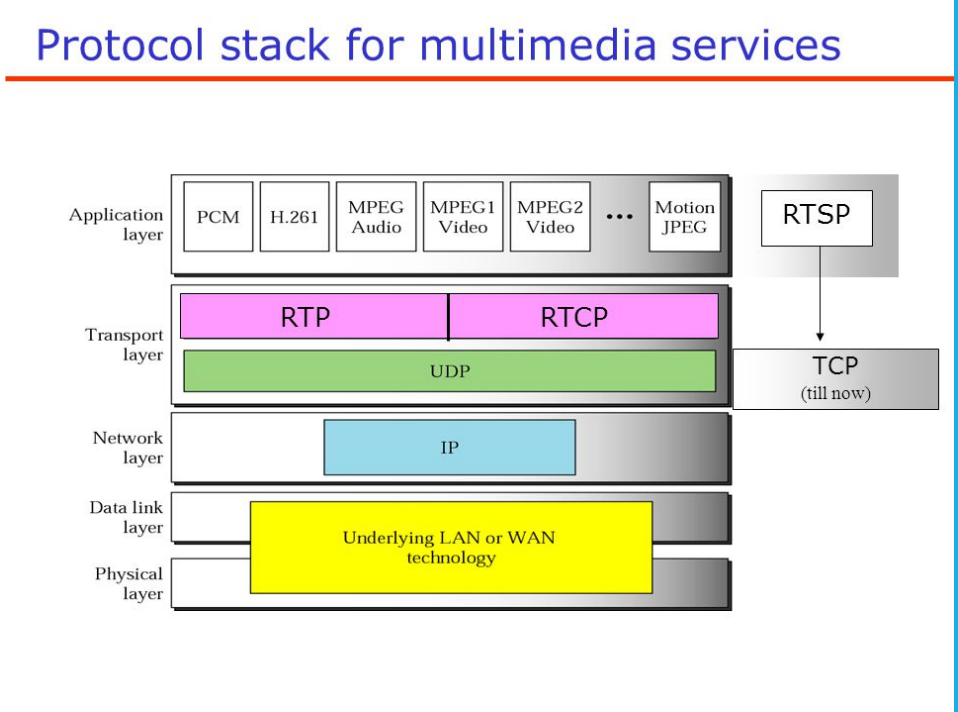
$$\boldsymbol{Atraso\ de\ transmissão} = \frac{\boldsymbol{Dimensão\ do\ pacote\ (bits)}}{\boldsymbol{Velocidade\ de\ transmissão\ (bps)}}$$

- O atraso de propagação depende da largura de banda do canal

$$\boldsymbol{Atraso\ de\ propagação} = \frac{\boldsymbol{Distância\ (Km)}}{\boldsymbol{Velocidade\ de\ propagação\ (Km/s)}}$$

Segurança e serviços

- Todas estas questões conduzem a pressão no desempenho da infraestrutura
- A diferenciação do tráfego e/ou a adoção de mecanismos para otimizar o desempenho torna-se por vezes necessário
- Entre os protocolos possíveis destaca-se o ***Real Time Transport Protocol*** (**RTP**)
  - RFC 3550 na sua definição mais recente
  - Orientado ao transporte de tráfego em tempo real, como vídeo, áudio, etc.
  - Não faz tratamento de erros nem controlo de fluxo
  - Opera em conjunto com o ***Real Time Control Protocol*** (**RTCP**) que controla o serviço e verifica a sua qualidade
- Enquanto o RTP opera sobre UDP, o ***Real Time Streaming Protocol*** (**RTSP**), RFC 2326, opera sobre TCP e permite o controlo do streaming, possibilitando pausá-lo, avançar, recuar, etc.



Fonte: PBXDom

Segurança e serviços

- Mas nem sempre a pressão é sobre tráfego interativo, e há o atraso de propagação ao longo do percurso atravessado, como já visto
- E esse pode ser melhorado, se possível através da redução do tamanho dos pacotes e/ou pela alteração do processamento dos pacotes nos dispositivos intermédios, privilegiando o tráfego considerado prioritário
- Qualquer tráfego contém um ***overhead*** provocado pelo encapsulamento ao longo do modelo TCP

$$\boldsymbol{overhead} = \frac{\boldsymbol{Informação\ total\ transmitida\ - Informação\ útil}}{\boldsymbol{Informação\ total\ transmitida}}$$

- Que por sua vez prejudica a **eficiência**

$$\boldsymbol{Eficiência} = \frac{\boldsymbol{Informação\ útil}}{\boldsymbol{Informação\ total\ transmitida}}$$

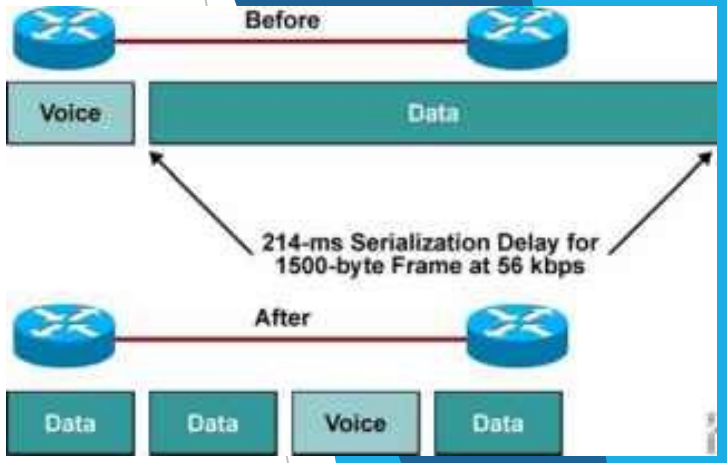
Segurança e serviços

Protocol headers	Total header size (bytes)	Min. compressed header size (bytes)	Compression gain (%)
IP4/TCP	40	4	90
IP4/UDP	28	1	96.4
IP4/UDP/RTP	40	1	97.5
IP6/TCP	60	4	93.3
IP6/UDP	48	3	93.75
IP6/UDP/RTP	60	3	95

Fonte: EFFNET AB

- Os cabeçalhos IP, RTP e TCP representam por vezes um peso significativo na eficiência e *overhead*
- Alguns mecanismos de compressão de cabeçalhos foram desenvolvidos permitindo uma melhoria razoável
- Destacam-se entre estes
  - **CTCP** (RFC 1144) – compressão de cabeçalhos IPv4 e TCP
  - **IPHC** (RFC 2507) – compressão de cabeçalhos IPv4, IPv6, UDP, TCP e RTP
  - **CRTP** (RFC 2508) – compressão de cabeçalhos IPv4, IPv6, UDP e RTP
  - **ROHC** (RFC 3095) – compressão de cabeçalhos IPv4, IPv6, ESP (que abordaremos mais tarde)
- A aplicação destes mecanismos é mais apropriada em ligações de baixo débito, podendo ser aplicados entre os extremos da ligação ou entre nós intermédios, de forma transparente para os nós finais

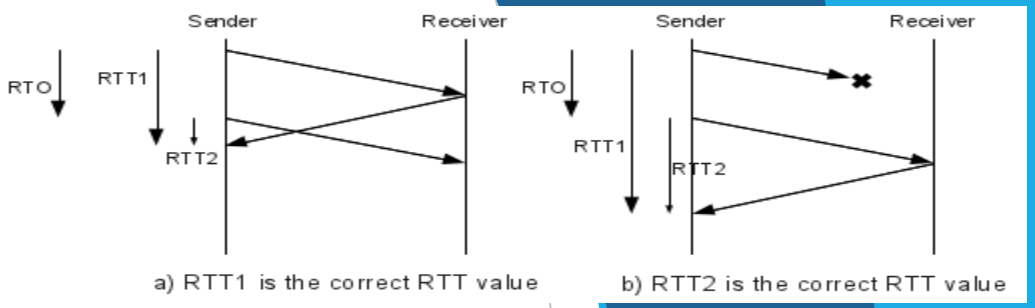
Segurança e serviços



Fonte: Cisco

- Mantendo-se o sistema FIFO nos nós que delimitam a ligação de baixo débito, a possibilidade de o tráfego pretendido ficar a aguardar que um outro tráfego de dimensão elevada atravesse o canal provoca um atraso que pode não ser razoável
- O ***Link Fragmentation and Interleaving*** (**LFI**) segmenta os pacotes grandes permitindo desta forma um atraso menor aos pacotes prioritários
- Esta técnica pode ser aplicada em nós intermédios de forma transparente para os nós finais

Segurança e serviços



Fonte: Everything about nothing

- Outra técnica possível para reduzir os atrasos é o controlo do congestionamento do TCP
- Este protocolo funciona como sabemos em modo ***stop & wait*** ou em **janela deslizante**
- No modo ***stop & wait*** o emissor envia um pacote e aguarda confirmação de receção (**ACK**) pelo recetor, aguardando um tempo máximo denominado ***Retransmission TimeOut*** (**RTO**)
- No modo **janela deslizante** o tamanho da janela é definido pelo recetor embora sendo ajustado durante a comunicação
- De acordo com a distância os nós e à ocupação dos canais intermédios, o **RTO** é também ajustado, medindo os tempos de demora entre a emissão e a receção do **ACK** – o ***Round Trip Time*** (**RTT**)
- Após algumas (poucas) transmissões o emissor calcula o **RTO** inicial como

$$RTO = 3\ x\ RTT$$

Segurança e serviços

- A janela deslizante pode provocar congestionamento no canal pois atrasos na confirmação da receção levarão ao reenvio dos pacotes considerados perdidos, pelo que um mecanismo adicional de controlo de fluxo foi definido
- A RFC 5681 define quatro algoritmos interligados de controlo do congestionamento do TCP
- O início da transmissão num canal com condições desconhecidas requer ao TCP que identifique as suas capacidades, evitando desta forma o seu congestionamento
- De igual forma, após detetar um congestionamento é necessário identificar se as condições de propagação foram alteradas
- Para tal define alguns parâmetros
  - ***Congestion window*** (**cwnd**) e ***Receiver window*** (**rwnd**)
  - ***Initial Window*** (IW)
  - ***Sender Maximum Segment Size*** (SMSS)
  - ***Slow start threshold*** (**ssthresh**) utilizado para determinar se o algoritmo de **slow start** ou de prevenção de congestionamento é utilizado para controlar a transmissão

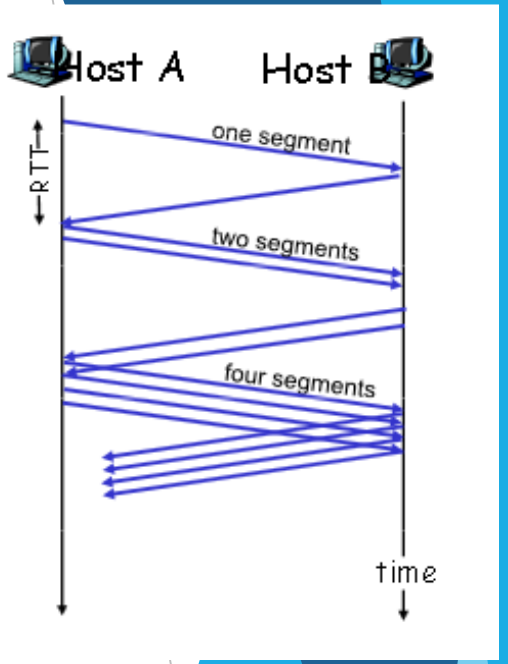
Segurança e serviços

- Neste modo a indicação inicial do tamanho máximo da janela é anunciado pelo recetor durante o estabelecimento da conexão (campo de 16 bits no cabeçalho TCP)
- O **cwnd** representa o limite de dados pendentes que o emissor pode transmitir antes de receber um **ACK**
- O **rwnd** representa o limite de dados pendentes que o recetor pode receber
- O mínimo destes dois valores, **cwnd** e **rwnd**, define a dimensão da janela
- O valor inicial do **IW** e de **cwnd** é definido como
  - Se **SMSS > 2190** bytes **IW = 2 x SMSS** e menor ou igual a 2 segmentos
  - Se **1095 bytes < SMSS <= 2190** bytes **IW = 3 x SMSS** e menor ou igual a 3 segmentos
  - Se **SMSS <= 1095** bytes **IW = 4 x SMSS** e menor ou igual a 4 segmentos

Segurança e serviços

- O valor inicial de **ssthresh** é arbitrariamente alto, mas deve ser reduzido para evitar congestionamento de tráfego
  - Pretende-se que sejam as condições da rede a definir o limite da janela e não as características do emissor ou recetor
- Se **cwnd < ssthresh** a rede encontra-se em modo **slow start**
- Mas se **cwnd > ssthresh** o controlo de congestionamento entra em ação
- No modo **slow start** o valor de **cwnd** é incrementado até ao máximo de **SMSS** para cada **ACK** recebido
- Quando **cwnd > ssthresh** ou é detetado congestionamento a transmissão sai do modo **slow start**
- No modo de controlo de congestionamento **cwnd** é incrementado em 1 segmento por **RTT**

Segurança e serviços



Fonte: Simon S. Lam

- O modo **slow start** traduz-se na prática pela verificação da largura de banda do canal de comunicação que pode ser aplicada na comunicação
  - O ajuste de **cwnd** é exponencial
- Caso ocorra a perda de dados **ssthresh** é ajustado para **50%** do **cwnd** anterior à perda
  - O que por sua vez vai implicar menor dimensão da janela, penalizando a comunicação
- Para tentar evitar penalizações a RFC 5681 define também o **Fast Retransmit / Fast Recovery**
- O recetor deve enviar um **ACK** duplicado se recebe um segmento fora de ordem
  - O que leva o emissor a assumir que há problemas na rede

Segurança e serviços

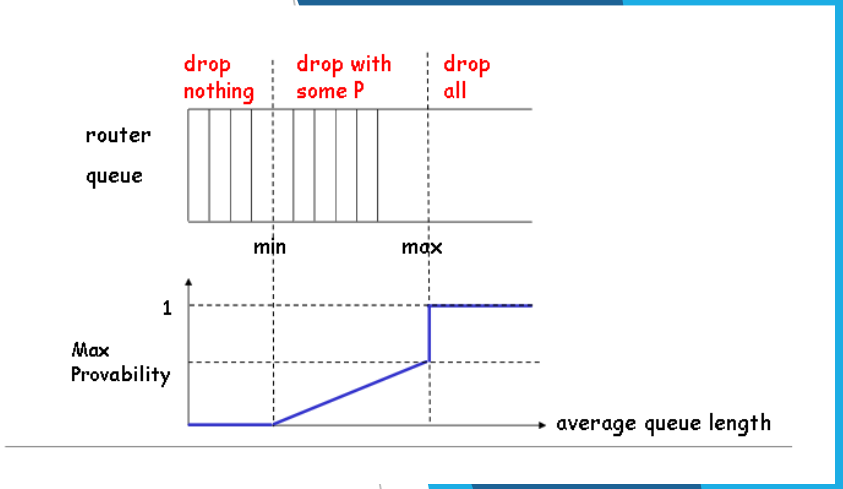
- Após 3 duplos **ACK** **cwnd** é ajustado para **50%** do valor atual e retoma-se o processo **slow start** embora com um aumento linear do **cwnd**
- Mas se ocorrer um **RTO, cwnd** é ajustado para o valor de **SMSS** e o processo **slow start** é retomado como inicialmente

- O modo **slow start** pode ser interpretado como uma variação da **janela deslizante** com o benefício de permitir um ajuste da dimensão da janela tendo em conta o emissor, o recetor e as condições da rede

Segurança e serviços

- O controlo de congestionamento pode tirar partido de mais mecanismos
- A característica habitual das filas de processamento dos ativos de comunicação é serem grandes; infelizmente pode implicar uma latência elevada e levarem o TCP a assumir que há um congestionamento e reverter para o mecanismo de **slow start**
- Quando a fila de processamento de um dispositivo fica cheia os pacotes seguintes que chegam são descartados – processo conhecido por ***tail-drop*** – que representa um peso importante nas comunicações
- O algoritmo ***Random Early Detection*** (**RED**) pode ser aplicado nos nós atravessados para evitar o ***tail-drop***

Segurança e serviços



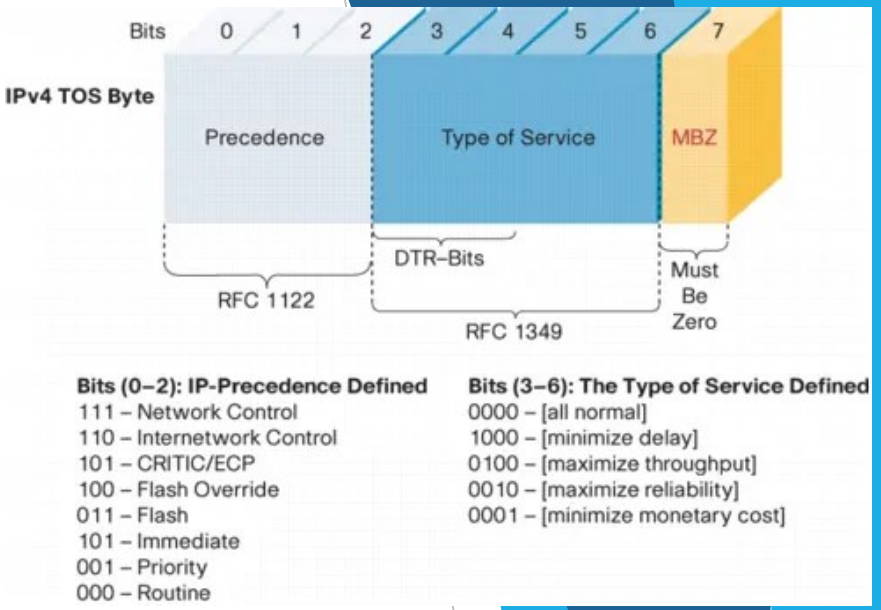
Fonte: Jin Hyin Ju

- O **RED** mantém dois valores que definem
  - A ocupação da fila (Min_{th}) a partir da qual descarta novos pacotes com probabilidade ***p*** crescente
  - A ocupação da fila (Max_{th}) a partir da qual todos os novos pacotes são descartados

Segurança e serviços

- O cabeçalho IP contém campos que permitem diferenciar o tráfego
  - No IPv4 o campo de 8 bits ***Type of Service*** (**TOS**)
  - No IPv6 o campo de 8 bits ***Traffic Class*** (**TC**)
- Estes campos permitem definir a prioridade do tráfego por forma a que os nós intermédios possam tratá-lo de forma diferenciada
- Em dispositivos mais antigos estes campos utilizavam apenas os 3 bits mais significativos representando o ***IP Precedence***
  - Definindo desta forma 8 (oito) classificações distintas

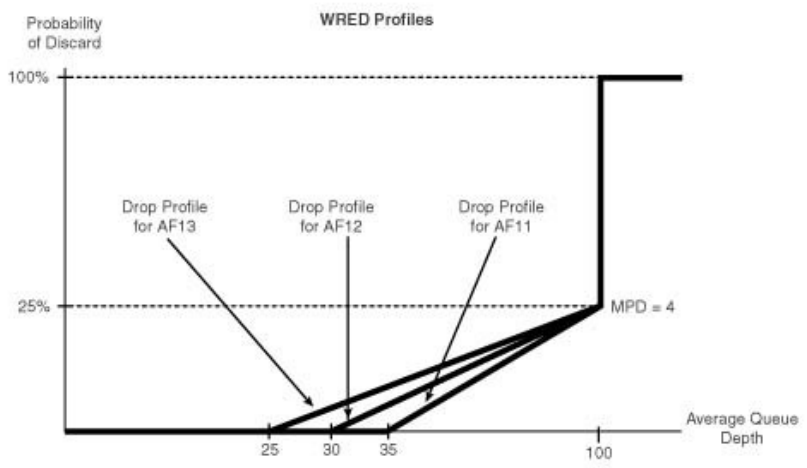
Segurança e serviços



Fonte: Cisco

- Posteriormente todos os bits passaram a ser utilizados sob a designação ***Differentiated Service*** (**DS**) também conhecido por ***DiffServ***
- Os 6 (seis) bits mais significativos são designados por ***Differentiated Service Code Point*** (**DSCP**) enquanto os 2 (dois) menos significativos são designados por ***Explicit Congestion Notification*** (**ECN**)
- No **DSCP** estão incluídos pois o **IP Precedence**, o que permite compatibilidade com nós antigos eventualmente ainda existentes no percurso
- Caso algum dispositivo atravessado não suporte **DS** o **DSCP** é ignorado nesse dispositivo mas mantém-se nos demais

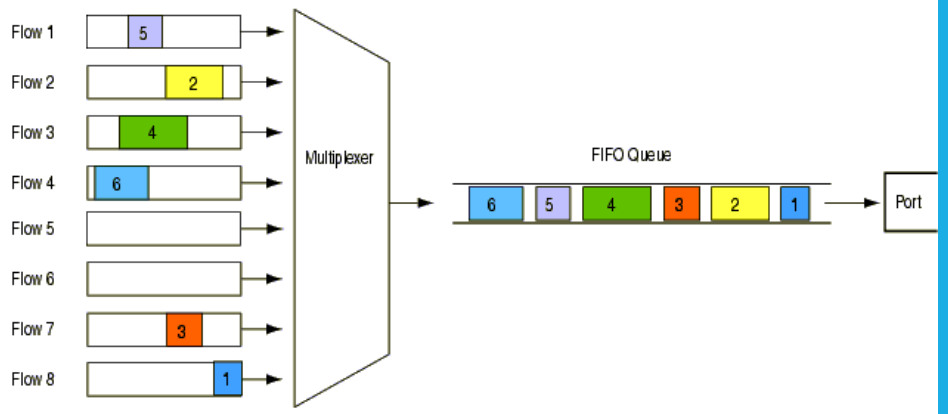
Segurança e serviços



Fonte: Cisco

- O **RED** tem o inconveniente de descartar pacotes de forma aleatória
- Em alternativa pode-se usar o ***Weight Random Early Detection*** (**WRED**) que opera segundo o mesmo princípio mas definindo prioridades de tipo de tráfego e um limite distinto para cada prioridade a partir do qual descarta pacotes de acordo com a ocupação da sua fila de processamento
- Como se observa na figura, qualquer pacote tem uma probabilidade de ser descartado mas a partir de percentagens de ocupação da fila distintos
- E, no limite, todos os pacotes poderão ser descartados quando a fila se apresentar completamente cheia

Segurança e serviços

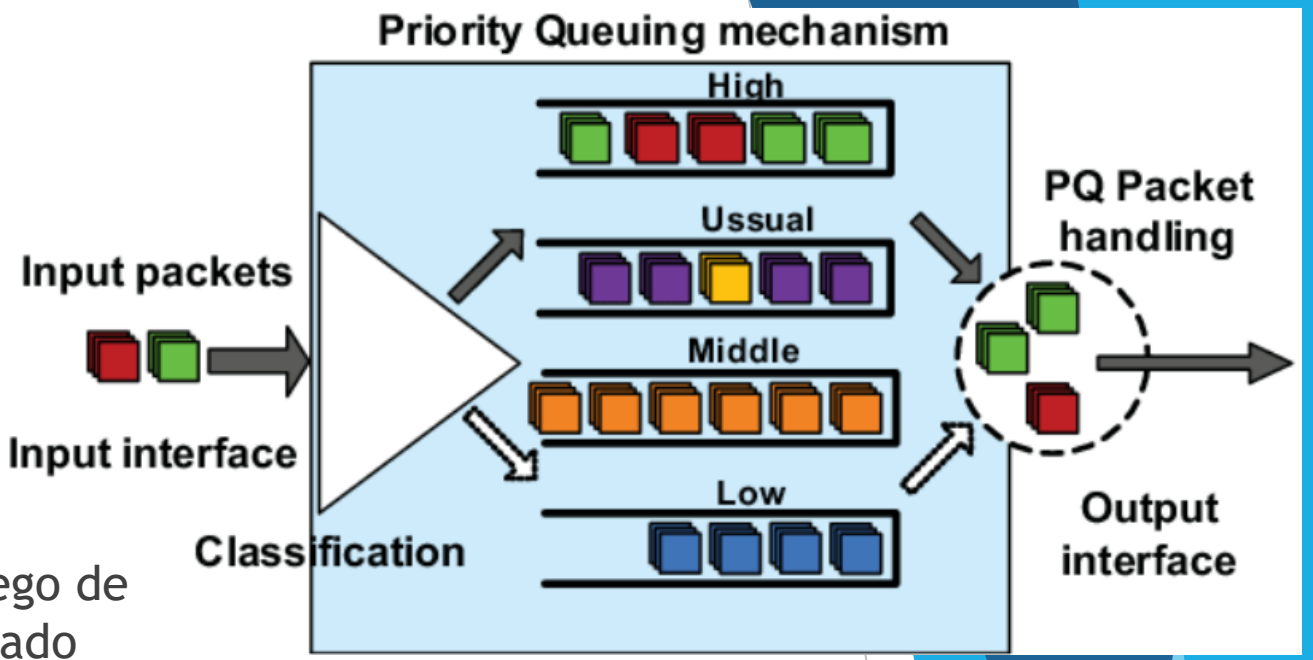


Fonte: Softwareopal

- Com base no **DSCP** é possível segmentar os pacotes de acordo com a sua importância, evitando deste modo o comportamento FIFO por omissão
- Contudo, os dispositivos atravessados pelos pacotes tratam por omissão todo o tráfego recebido de igual forma, pelo que para se tirar partido do **DCSP** é necessário configurá-los para ***Qualidade de Serviço*** (**QoS**)
- O **QoS** tem dois métodos, ***Soft QoS*** e ***Hard QoS***, ambos criando filas (*queles*) distintas e paralelas para colocação do tráfego de acordo com a sua classificação
  - O **Soft QoS** partilha os recursos de acordo com os tipos de tráfego não reservando parte da capacidade exclusivamente para nenhum
  - O **Hard QoS** reserva recursos de forma estática para tipos de tráfego definidos

Segurança e serviços

- **Soft QoS** - ***Priority Queuing*** **(PQ)**
- Sempre que há tráfego numa fila de maior prioridade é sempre processado antes do tráfego de prioridade imediatamente inferior ser processado
- Isto implica que tráfego de menor prioridade pode no limite nunca ser processado
- O número de filas é definido na configuração do ativo e a alteração das filas implica normalmente o reinício do ativo



Fonte: ResearchGate

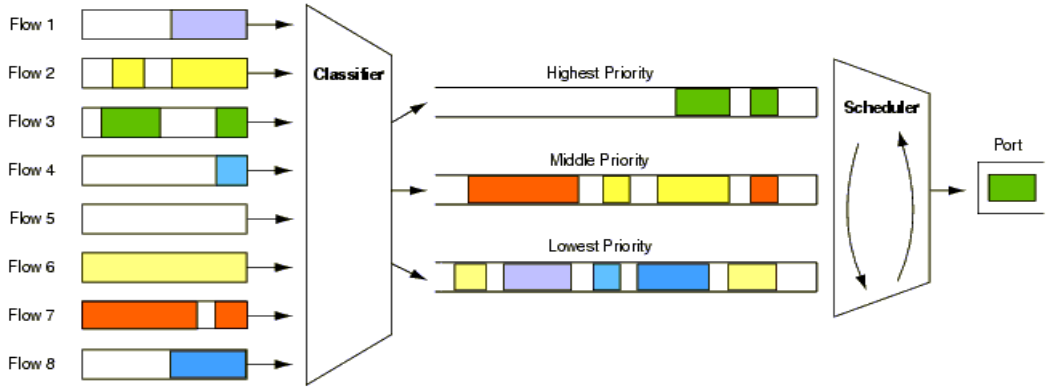
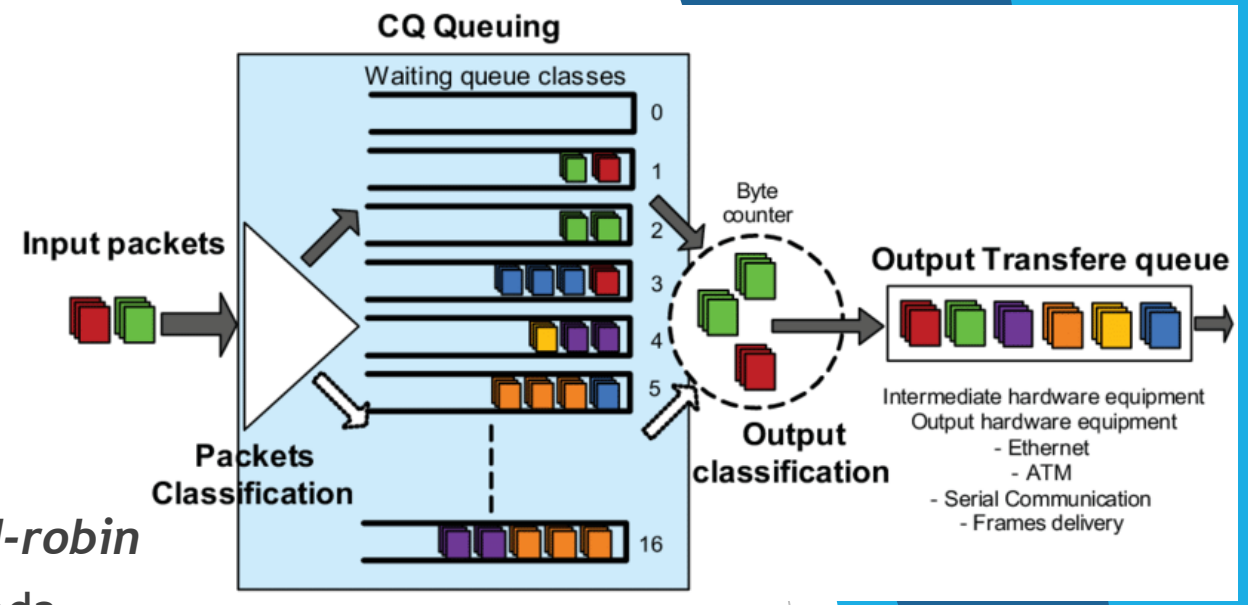


Figure 2.3.1

Fonte: Softwareopal

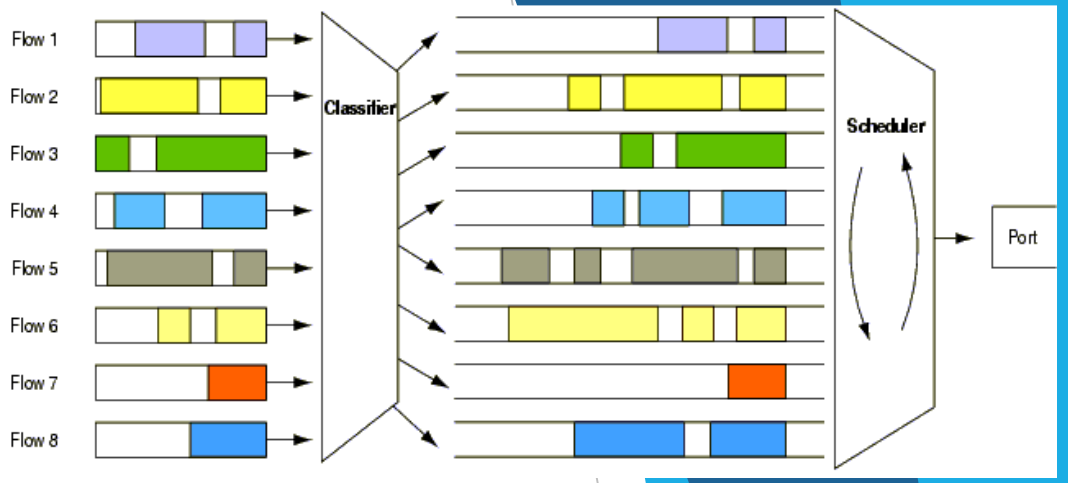
Segurança e serviços



Fonte: ResearchGate

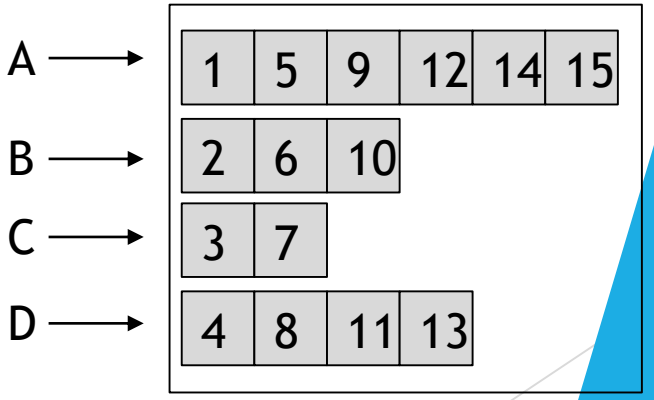
- **Soft QoS** - ***Custom Queuing*** (**CQ**)
- Processa pacotes de cada fila num sistema ***round-robin***
- Todas as filas são percorridas, processando em cada uma delas pacotes até o número de bytes que foi definido seja excedido
- O **CQ** permite uma distribuição mais razoável da capacidade de saída, privilegiando na mesma o tráfego mais prioritário mas sem que isso implique a ausência de processamento do menos prioritário

Segurança e serviços



Source: helisetalem.com

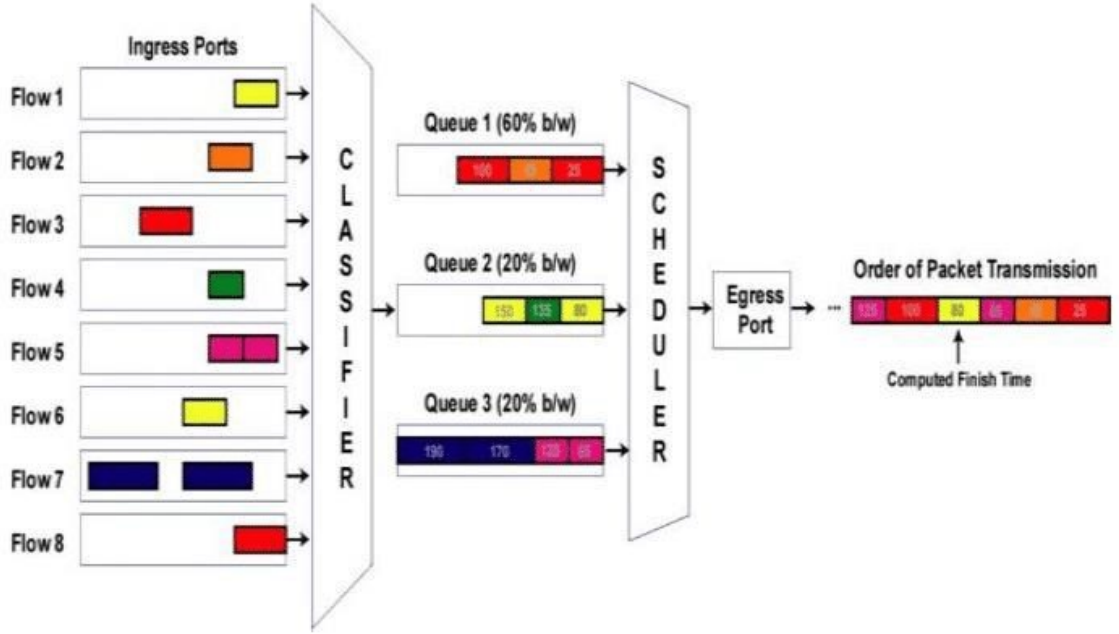
- **Soft QoS** - ***Fair Queuing*** **(FQ)**
- Processa pacotes de cada fila num sistema ***round-robin***
- O número de filas criadas é dinâmico, dividindo de igual forma pelos diferentes tipos de tráfego em processamento
- Havendo alteração do número de tipos de tráfego, o número de filas altera-se automaticamente
- O **FQ** adapta-se de forma dinâmica aos fluxos distintos
- Em cada fila é processado um pacote de cada vez
- Assumindo que num determinado instante há **4** filas **A** a **D** e diferentes fluxos de tráfego em cada uma, teremos



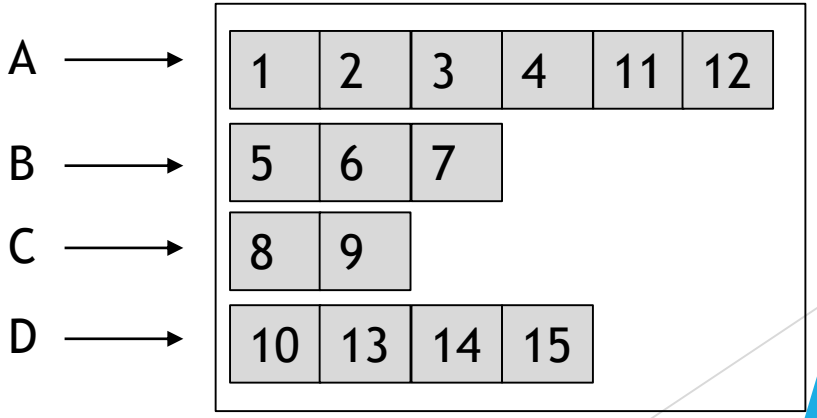
Queue	End time
A	15
B	10
C	7
D	13

Security and services

- **Soft QoS** - ***Weight Fair Queuing*** **(WFQ)**
- Similar ao **FQ** mas a cada fila é atribuída uma prioridade
- A capacidade do canal de saída não é distribuída de forma igual pelas filas existentes, mas como a média pesada das filas
- Assumindo que num determinado momento há **4** filas criadas com pesos **A=4**, **B=3**, **C=2**, **D=1** e diferentes fluxos de tráfego para cada um deles, teremos



Fonte: ResearchGate

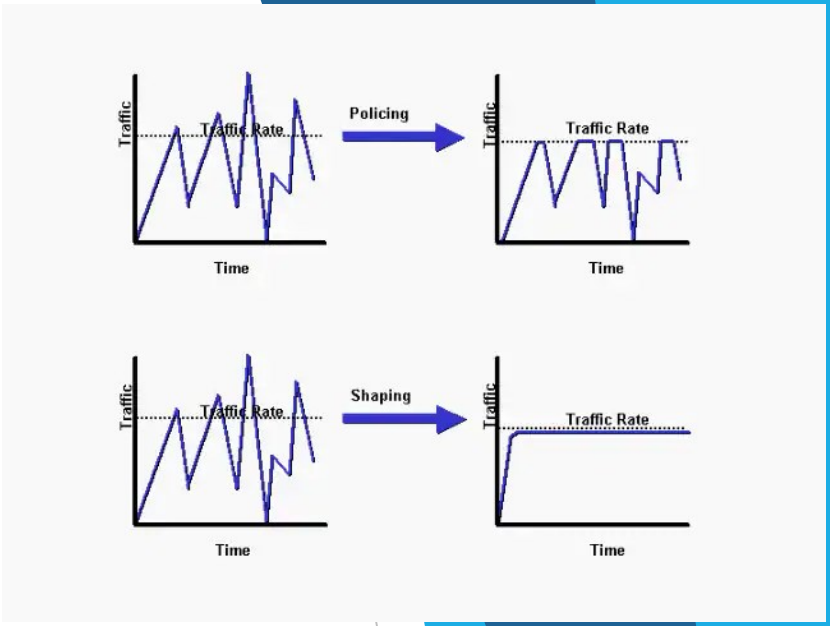


Fila	Tempo final
A	12
B	7
C	9
D	15

Segurança e serviços

- O **Hard QoS** reserva recursos de forma estática para tipos de tráfego
- A sua principal desvantagem é o não aproveitamento das reservas configuradas caso não haja tráfego desse tipo, o que implica desperdício de recursos
  - Por exemplo, se reservarmos 70% da capacidade para tráfego HTTP, mesmo que esse tipo de tráfego não exista em determinado momento os outros tipos terão apenas 30% disponíveis
- Em contrapartida, o tráfego para o qual foi reservado recursos não sofre atrasos derivados de outro tipo de tráfego
- Em caso de excesso de tráfego, pode sempre ocorrer o esgotamento da largura de banda assignada para um determinado tráfego
- Para minimizar o impacto dessa situação, utiliza dois conceitos complementares que podem ser usados em simultâneo
  - ***Traffic Policing***
  - ***Traffic Shaping***

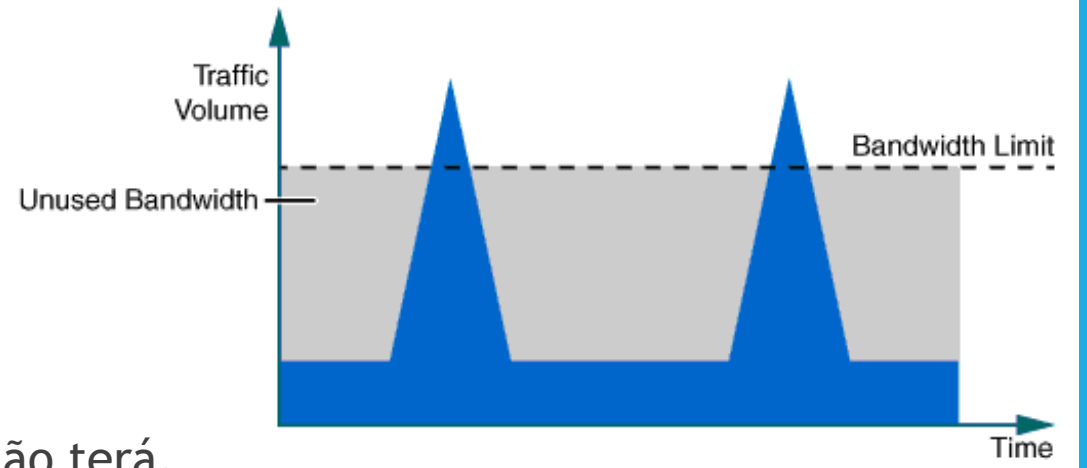
Segurança e serviços



Fonte: Cisco

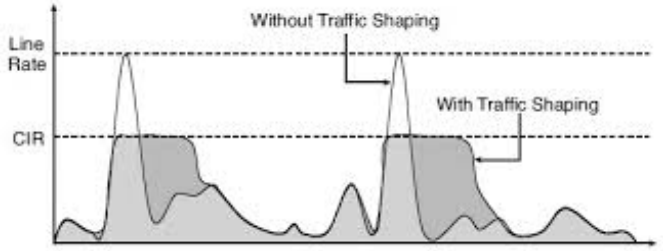
- O ***Traffic Policing*** consiste na eliminação pura e simples do tráfego que excede a capacidade que lhe está atribuída
- O ***Traffic Shaping*** consiste em reter o tráfego que excede a capacidade que lhe está atribuída, e retransmiti-lo quando o canal estiver mais livre
- A retransmissão é calendarizada após ter decorrido um determinado intervalo de tempo repetindo-se o processo até transmissão com sucesso
- Aplica-se contudo na mesma que se a capacidade de retenção de tráfego excede os recursos disponíveis, novo tráfego em excesso será descartado

Segurança e serviços

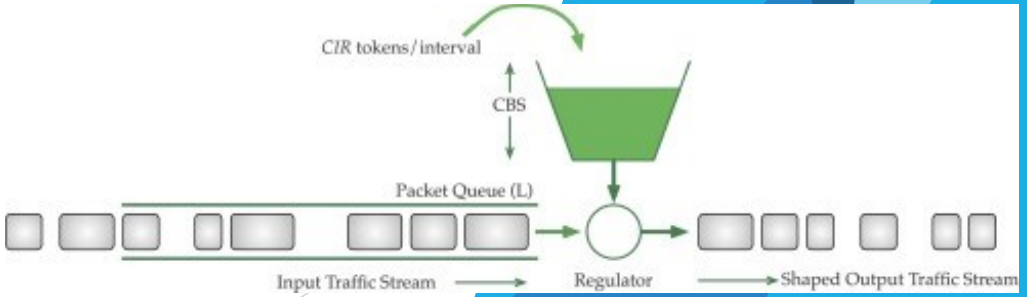


Fonte: Juniper.net

- O tráfego que atravessa o ativo onde é aplicado o **QoS** não terá, em termos médios, uma ocupação constante, havendo momentos de muito tráfego e outros de pouco tráfego
- Pode-se assim estimar uma ocupação média e, com base nela, ocupação em excesso
- Com recurso ao ***Traffic Shaping*** o número de pacotes que potencialmente seriam descartados é minimizado
  - Contudo, a taça da figura em baixo pode encher…

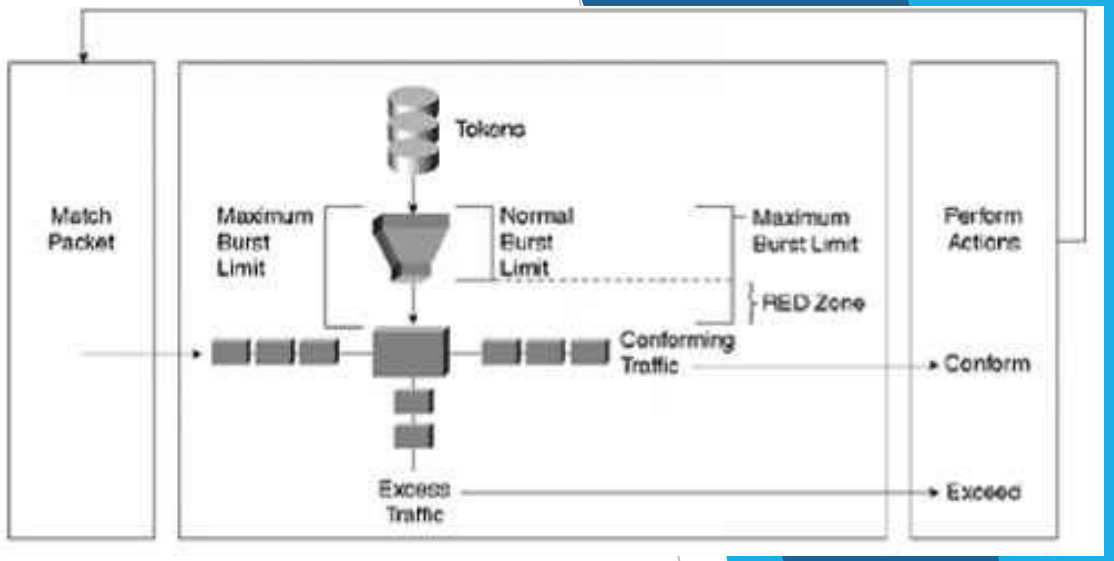


Fonte: Tutorzine.com



Fonte: Ronald. J.Vetter

Segurança e serviços



Fonte: Cisco

- A aplicação de técnicas de ***Traffic Policing*** e ***Traffic Shaping*** são dependentes das marcas dos ativos
  - Por exemplo, o ***Committed Access Rate*** (**CAR**) e o ***Token Bucket***, no ***Traffic Policing***
  - O ***Leaky Bucket*** e o ***Token Bucket*** no ***Traffic Shaping***

- A aplicação do ***Traffic Shaping*** implica a alteração em tempo real do campo ***DiffServ*** do pacote

Segurança e serviços

- O tráfego de forma global pode ser dividido entre *tradicionais* (de aplicações vulgares) ou *tempo-real* (de *streaming*)
- O tradicional pode habitualmente sofrer atrasos e perdas sem um prejuízo significativo
  - Desde, claro, que não tenha consequências no **SLA**
- Mas as aplicações de tempo-real são menos tolerantes aos atrasos, variações (*jitter*) e às perdas
- Para estas aplicações o ***Hard QoS*** que contemple os mecanismos de ***Traffic Policing*** e ***Traffic Shaping*** pode não ser apropriado

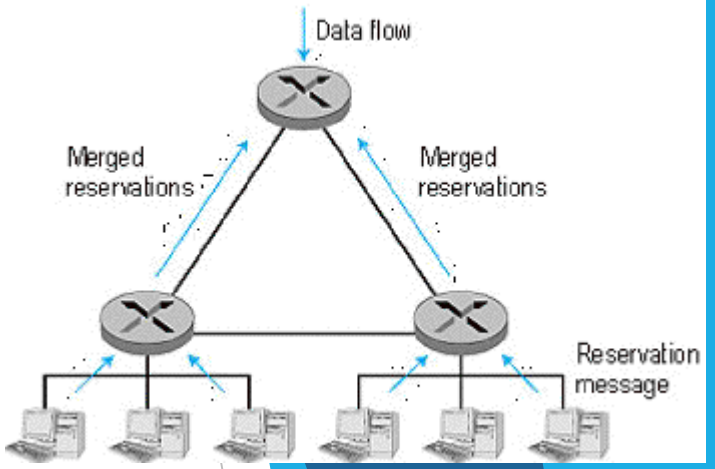
Segurança e serviços

- Daí uma abordagem de ***Hard QoS*** distinta ao ***DiffServ*** (e por vezes considerada oposta): o ***Integrated Services*** (**IntServ**), motivado por necessidades de aplicações de tempo real
- O ***IntServ*** é orientado ao *multicast* e foca-se na reserva de recursos e mecanismos de controlo de acesso dos pacotes ao longo do canal que interliga o emissor e o recetor
- Enquanto o *DiffServ* assegura uma melhor escalabilidade de QoS, o *IntServ* providencia um mecanismo mais restrito para tempo real

Segurança e serviços

- O *IntServ* necessita de várias funções ao longo dos dispositivos atravessados pelos pacotes
  - Controlo de admissão para determinar se um novo fluxo pode assegurar o QoS pretendido sem afetar reservas já existentes
  - Classificação para reconhecer os pacotes que necessitam de níveis específicos de QoS
  - Políticas para agir (incluindo descartar pacotes) quando não houver conformidade com as características especificadas
  - Enfileiramento e agendamento para tratar os pacotes de acordo com as prioridades pretendidas
- Um protocolo que utiliza esta abordagem é o ***Resource ReSerVation Protocol*** (**RSVP**)

Segurança e serviços



Fonte: ipsit.bu.edu

- O **RSVP** é um protocolo orientado ao recetor
- É o nó de destino que contacta os nós ao longo do percurso para solicitar a reserva de recursos necessária
- O **RSVP** não é um protocolo de encaminhamento, consultando e tirando partido das tabelas de encaminhamento ao longo do percurso para obter as rotas necessárias, quer em *unicast* quer em *multicast*
- Trabalha em modo *simplex*, isto é, as reservas são asseguradas para um sentido só do fluxo
- Um pedido de reserva **RSVP** pode ter dois tipos, designados *styles*
  - *Distinct* em que é solicitada uma reserva para um tráfego exclusivo
  - *Shared* em que fluxos distintos (inclusive de emissor e recetor) partilham a mesma reserva

Segurança e serviços

- As reservas podem necessitar de ser alteradas, por exemplo por a tabela de encaminhamento ser alterada ou mais tráfego necessitar de reserva
- Cada reserva tem um tempo de vida ao fim do qual deve ser renovada
  - Por omissão, o tempo de vida das reservas é de 30 seg.
  - Este tempo pode ser alterado mas tendo presente que se valores mais baixos vão otimizar a adaptação da rede, vão também aumentar o *overhead* inerente ao RSVP
- Os nós ao longo do percurso podem suportar RSVP ou não
  - Se não suportam, ignoram os pedidos de reserva e encaminham-nos para o nó seguinte
  - Se suportam, processam-nos, encaminham-nos para o nó seguinte e informam o emissor do pedido de reserva
- Os nós que suportam RSVP podem agregar pedidos recebidos que forem compatíveis
  - Por isso, a garantia da reserva solicitada é sempre parcial

Administração de Sistemas ASIST

Tópico 6

Túneis e protocolos de segurança

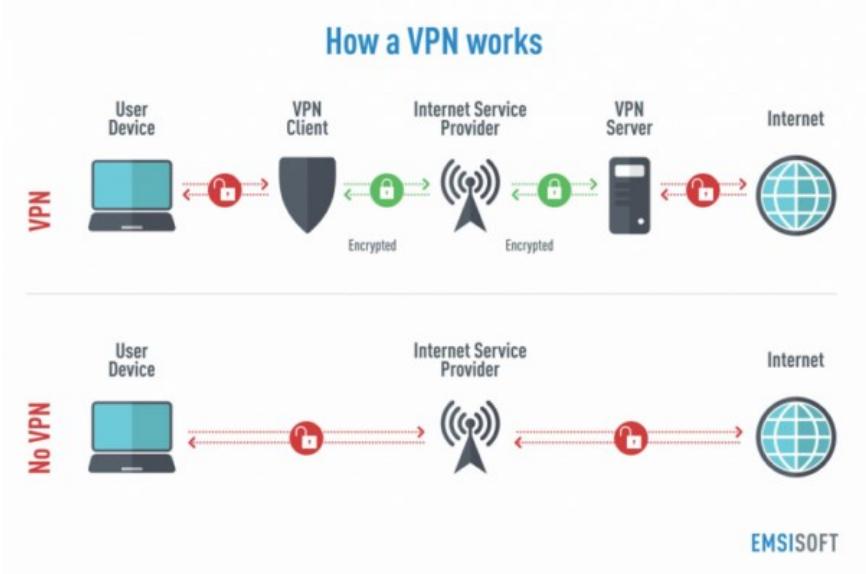
Pinto Leite, Jorge (jpl@isep.ipp.pt)

Túneis e protocolos de segurança

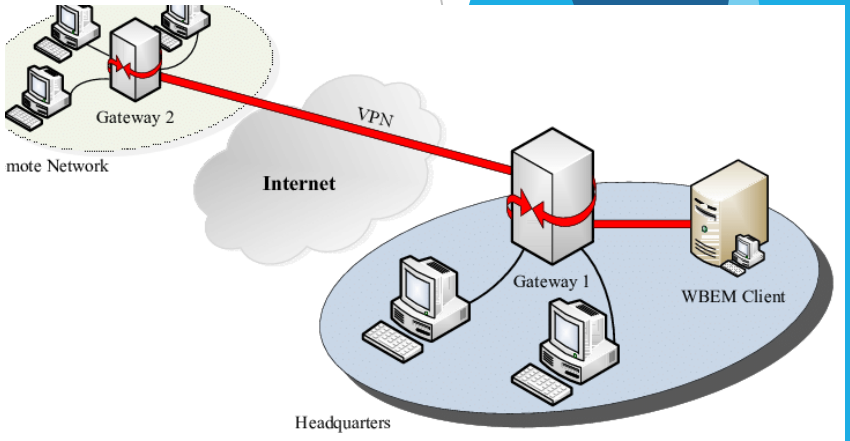
- A informação que circula na rede, seja interna ou externa, deve ser protegida se sensível
- A proteção pode ser providenciada através de uma ***Virtual Private Network*** (**VPN**) que aplica algoritmos criptográficos por forma a oferecer a confidencialidade e integridade
- A VPN pode (e deve) existir sempre que circule informação cuja segurança se pretende alcançar, quer em redes internas quer em externas
- O estabelecimento do túnel é assegurado pelos protocolos configurados e só após a sua criação é que a informação começa a ser trocada
- São assim uma forma potencialmente económica de assegurar a segurança da comunicação entre pontos distintos qualquer que seja o canal, seguro ou inseguro, entre eles
  - *Potencialmente* pois apenas alguns dos protocolos de VPN estão incluídos por omissão

Túneis e protocolos de segurança

- Uma VPN pode ser criada entre um posto e uma rede (HOST – LAN) ou entre redes (LAN – LAN)
- **HOST - LAN**
  - Exige criação pelo utilizador
  - No HOST é criada uma interface virtual que contém as características de ligação (endereço de destino, definições, protocolos permitidos, etc.)
  - Do lado da LAN, exige um sistema de destino (servidor ou *appliance*), configurado para esse serviço
- **LAN - LAN**
  - Configurada entre sistemas ou *appliances*, sendo transparente para o utilizador



Fonte: Nationalcybersecuritynews



Fonte: ResearchGate

Túneis e protocolos de segurança

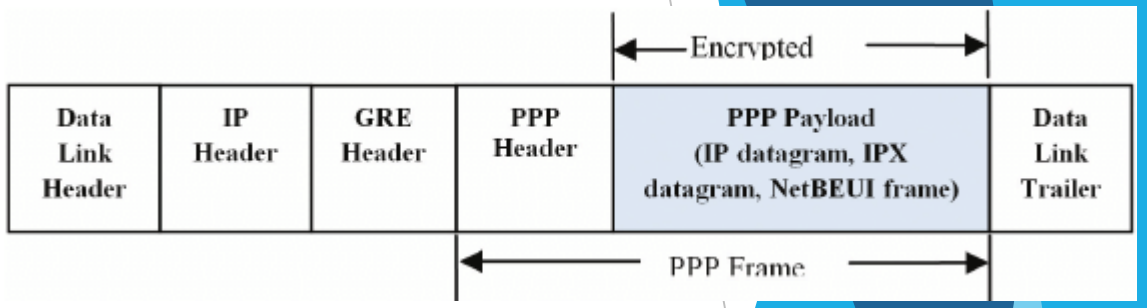
- A criação do túnel pode provocar constrangimentos na comunicação dado o valor do ***Maximum Transmission Unit*** (**MTU**)
  - Por omissão, é normalmente de 1500 bytes
- Com o *overhead* inerente à aplicação dos protocolos de túnel, pode implicar a necessidade da fragmentação nem sempre eficiente e por vezes até não suportada
- O ***Path MTU Discovery*** (**PMTUD**), RFC 4821, tira partido do ICMP e possibilita a deteção do máximo MTU que pode ser aplicado ao longo do túnel
- De notar que por vezes o protocolo ICMP pode estar bloqueado em *firewalls*, pelo que o uso de outras técnicas como o recurso a outros métodos, como o uso de protocolos do nível 4 do modelo OSI ou a alteração do ***Maximum Segment Size*** (**MSS**), estão contemplados na RFC

Túneis e protocolos de segurança

- Vários protocolos podem ser usados para a implementação da VPN, entre eles
  - *Point-to-Point Tunneling Protocol* (PPTP)
  - *Layer 2 Tunneling Protocol* (L2TP)
  - *Internet Protocol Security* (IPsec)
  - *Secure Socket Layer* (SSL)
  - *Transport Layer Security* (TLS)
  - *Secure Socket Tunneling Protocol* (SSTP)

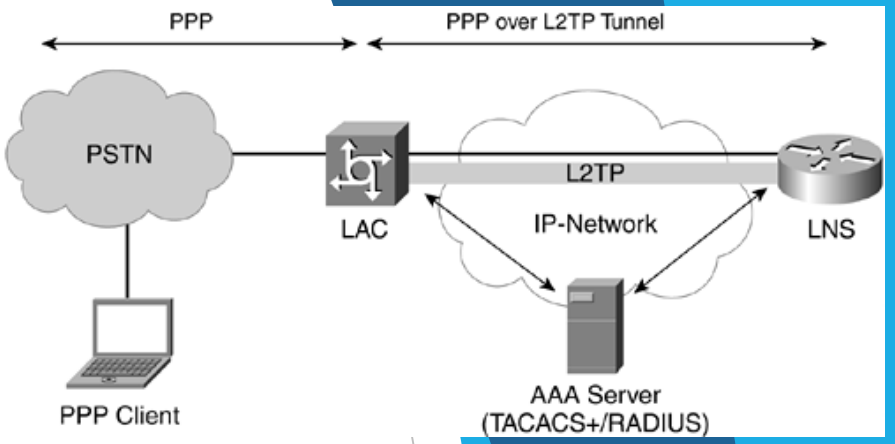
Túneis e protocolos de segurança

- ***Point-to-Point Tunneling Protocol*** (**PPTP**)
  - Baseado no ***Generic Routing Encapsulation*** (**GRE**) é um protocolo antigo
  - Nunca foi quebrado, contudo as suas implementações práticas (da Microsoft) apresentam vulnerabilidades, pelo que é considerado pouco seguro (depende da palavra-chave do utilizador)
  - Implica um baixo *overhead*
  - Cria uma sessão ***Point-to-Point Protocol*** (**PPP**) dentro do túnel criado
  - É simples de instalar e configurar, não exige certificados para operar, pelo que se tornou bastante popular nas ligações HOST - LAN



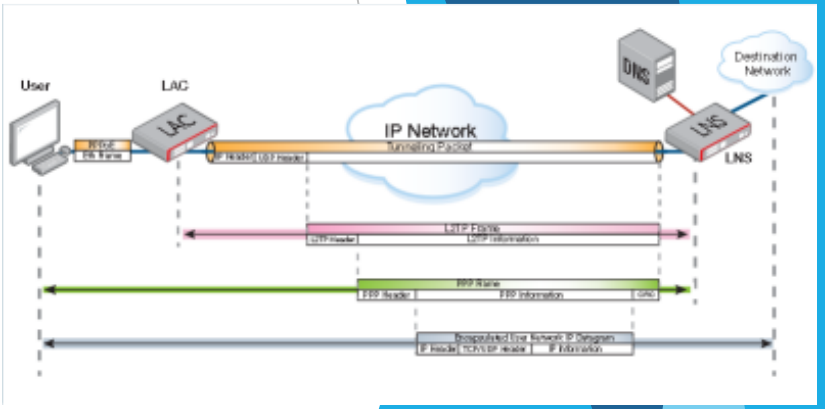
Fonte: ResearchGate

Túneis e protocolos de segurança

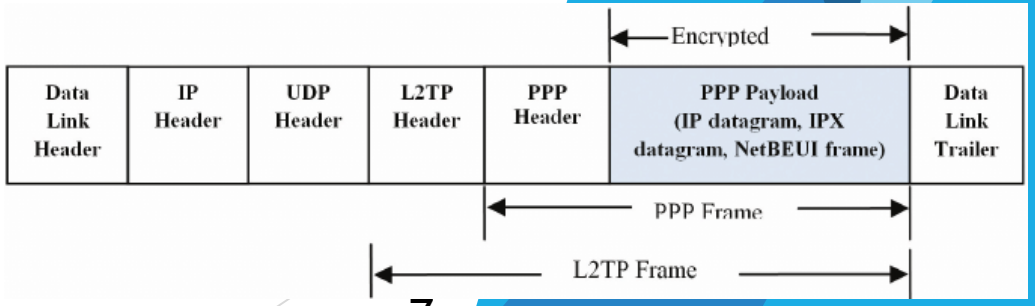


Fonte: ResearchGate

- ***Layer 2 Tunneling Protocol*** (**L2TP**)
  - Reúne as melhores características do **PPTP** e do ***Layer2 Forwarding*** (**L2F**), um protocolo desenvolvido pela Cisco
  - Contém as funcionalidades para implementação da VPN, contudo não inclui confidencialidade
    - A própria RFC sugere a implementação em conjunto com outro protocolo, o IPsec, para assegurar a confidencialidade
  - Os extremos da ligação podem operar como ***L2TP Access Concentrator*** (**LAC**) ou ***L2TP Network Server*** (**LNS**)
    - O **LAC** opera no nível 2
    - O **LNS** opera no nível 3, mas a comunicação é sempre no nível 2
  - Permite a ligação **LAC - LAC**, **LAC - LNS** ou **LNS - LNS**



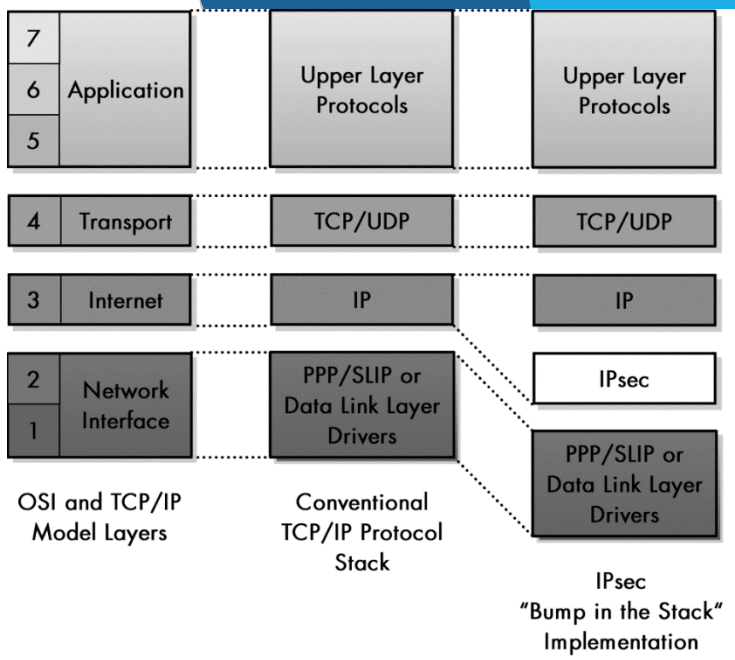
Fonte: Allied Telesis



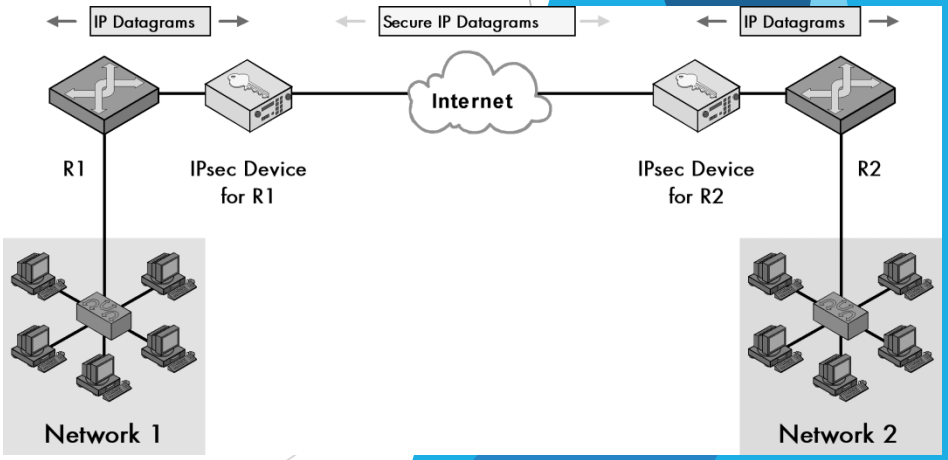
Fonte: ResearchGate

Túneis e protocolos de segurança

- ***Internet Protocol Security*** **(IPsec)**
  - Proporciona dois modos de funcionamento
    - **Transporte**
      - O cabeçalho do pacote original mantém-se
    - **Túnel**
      - O pacote original é encapsulado num novo pacote
  - Proporciona dois mecanismos de segurança que podem ser aplicados em conjunto
    - ***Authentication Header*** **(AH)**
    - ***Encapsulating Security Payload*** **(ESP)**
    - Se ambos os mecanismos forem aplicados, o AH precede sempre o ESP
  - Pode ser implementado em modo LAN – LAN (***Bump-in-the-wire*** **BITS**) ou HOST – LAN (***Bump-In-The-Stack*** **BITS**)

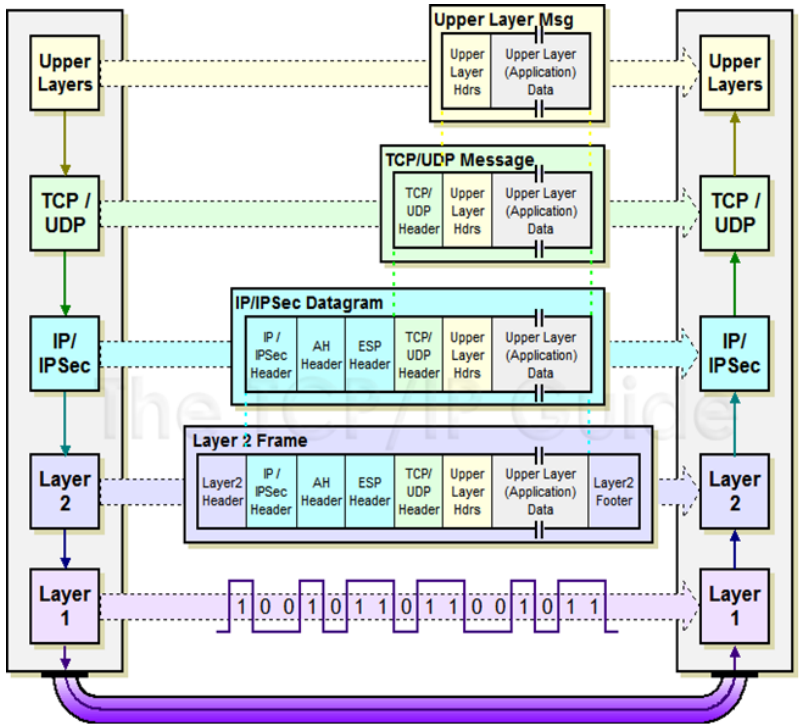


Fonte: Kozierok, C.M., *The TCP/IP Guide*

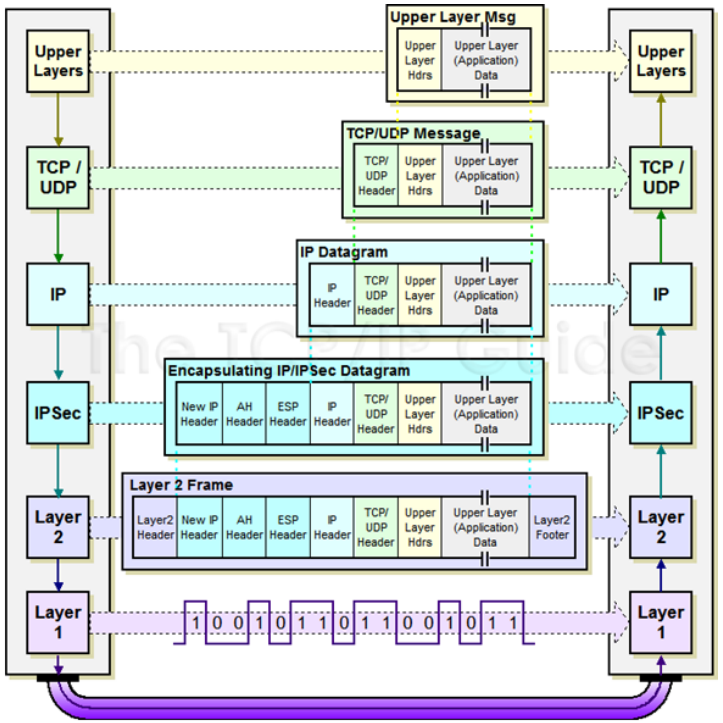


Fonte: Kozierok, C.M., *The TCP/IP Guide*

Túneis e protocolos de segurança



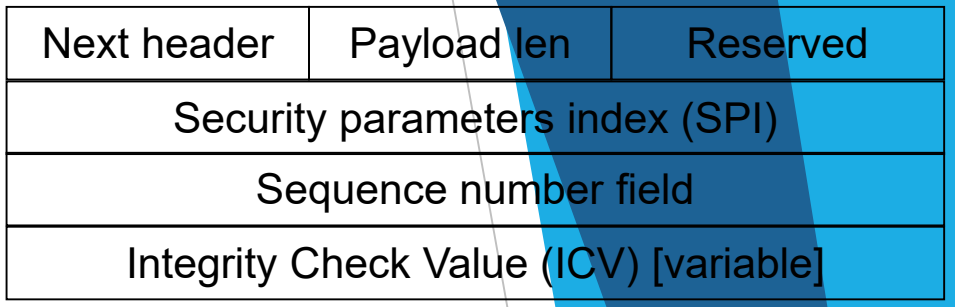
Transporte



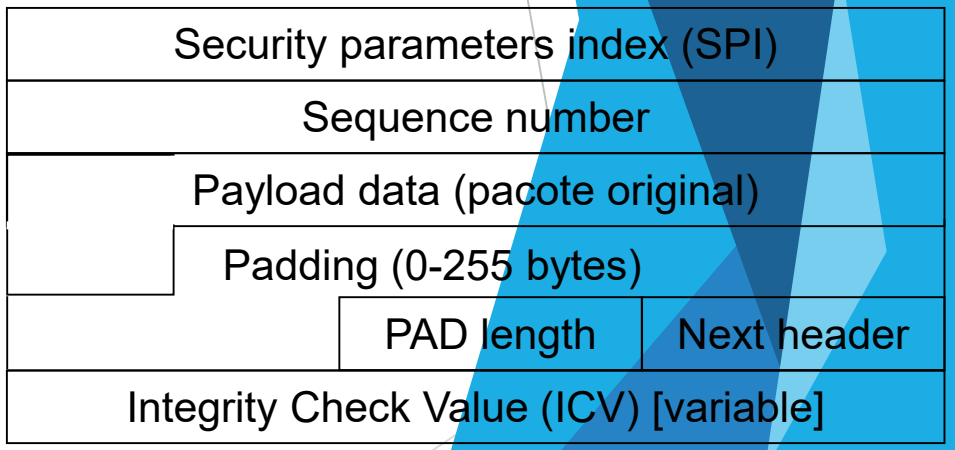
Túnel

Túneis e protocolos de segurança

- ***Internet Protocol Security*** **(IPsec)**
  - ***Authentication Header*** **(AH)**
    - Introduz no pacote um cabeçalho AH (protocolo 51)
    - Garante a integridade de todo o pacote (os *mutable fields* são ignorados)
    - Pode ainda garantir o não *replay* do pacote
  - ***Encapsulating Security Payload*** **(ESP)**
    - Introduz no pacote um cabeçalho ESP (protocolo 50)
    - Encapsula o pacote original num novo, incorporando-o no *payload*
    - Garante a confidencialidade dos dados no *payload* (onde se inclui o pacote original)
    - Pode ainda garantir a integridade do pacote
    - Introduz aleatoriamente pacotes *dummy* no canal para mascarar a ausência de tráfego
  - Se ambos são aplicados, AH precede ESP



Cabeçalho AH



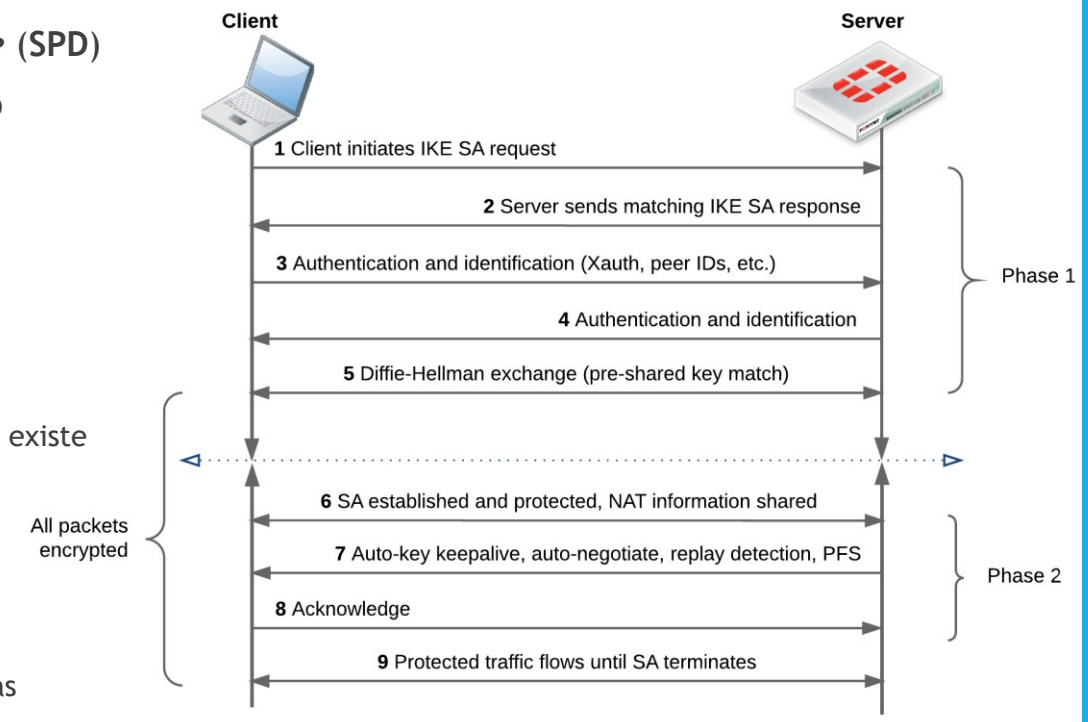
Cabeçalho ESP

Túneis e protocolos de segurança

- Para cada ligação e modo entre os extremos da ligação IPsec (*peers*) é criada uma ***Security Association*** (**SA**) unidirecional
  - Para possibilitar a comunicação nos dois sentidos, duas SA são criadas, uma para cada sentido
  - Se é aplicado em simultâneo o AH e o ESP, são criadas quatro SA
- Os algoritmos criptográficos aplicados aos mecanismos de segurança são negociados entre os disponíveis nos dois extremos e registados numa base de dados denominada ***Security Association Database*** (**SAD**)
- O ***Security Parameter Index*** (**SPI**) visível nos cabeçalhos do slide 10, identifica a entrada da SA na SAD e, por inerência, os algoritmos criptográficos aplicados
- As SA possuem um tempo de vida associado à sua duração e/ou ao volume de tráfego trocado

Túneis e protocolos de segurança

- A definição do tráfego a proteger é armazenado na ***Security Policy Database*** (**SPD**)
- Quando um pacote é enviado, o emissor procura na SPD se deve ser protegido
  - Se deve
    - Cria, caso não exista ainda, a **SA** e regista-a na **SAD**
    - Se a **SA** já existe, processa o pacote de acordo com as regras especificadas na **SAD**
  - Se não deve, o pacote é enviado sem proteção
- No recetor
  - Caso o pacote seja protegido, é pesquisada a **SAD** (pois a **SA** já foi criada pelo que já existe na **SAD** do recetor) para processar o pacote
    - Caso não exista, descarta o pacote
  - Caso o pacote não seja protegido, é processado normalmente
- O IPsec pode trabalhar com o ***Internet Key Exchange Protocol*** (**IKE**)
  - Nesse caso o **IKE** cria uma **SA** bidirecional dentro da qual as **SA** IPsec são encapsuladas
  - O **IKE** possibilita a gestão dos algoritmos aplicados à **SA**, alterando-os dinamicamente



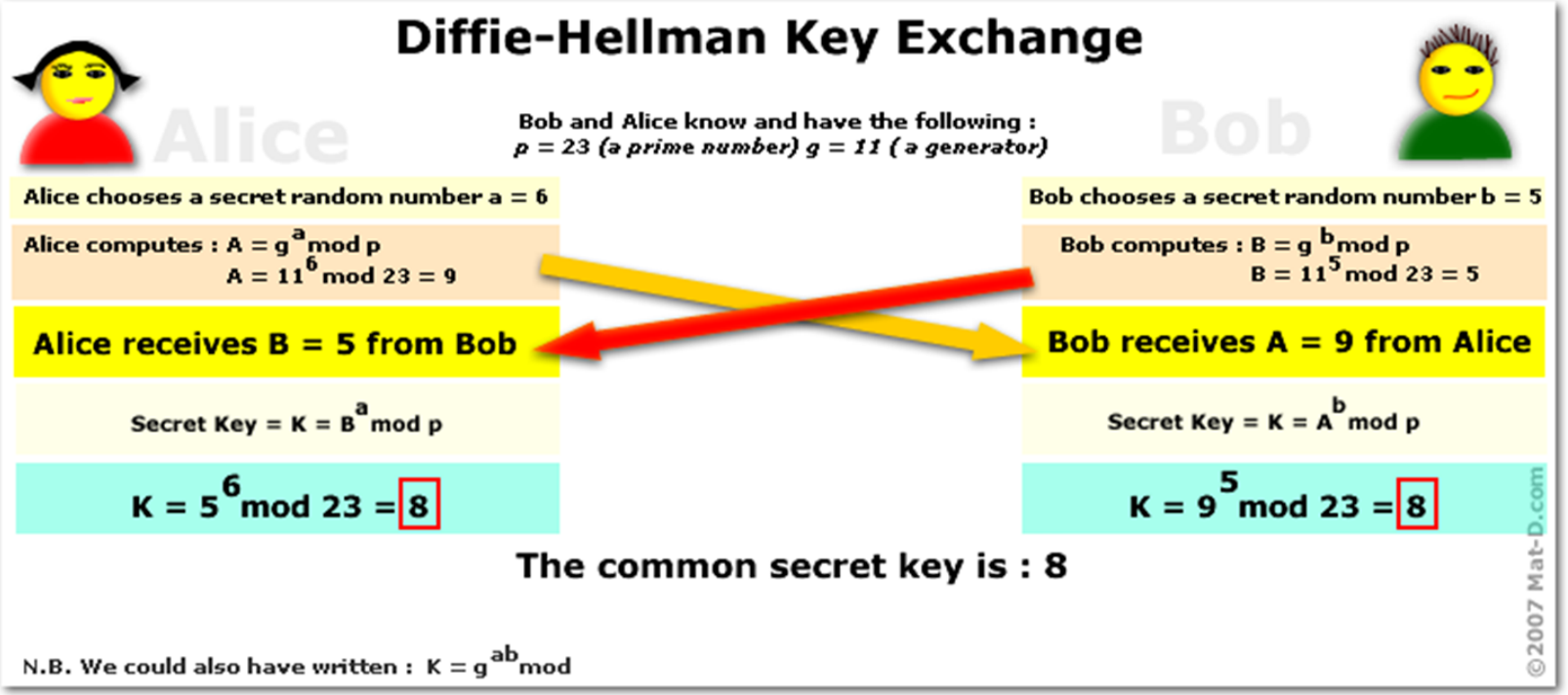
Fonte: Fortinet

Túneis e protocolos de segurança

- As vantagens do IPsec são várias
  - Autenticação (AH / ESP)
  - Anti-*replay* (opcional)
  - Confidencialidade (ESP)
  - Confidencialidade de fluxo de tráfego (ESP)
- Mas tem também inconvenientes
  - *Overhead* (AH / ESP)
  - Tempo de processamento (ESP)

Túneis e protocolos de segurança

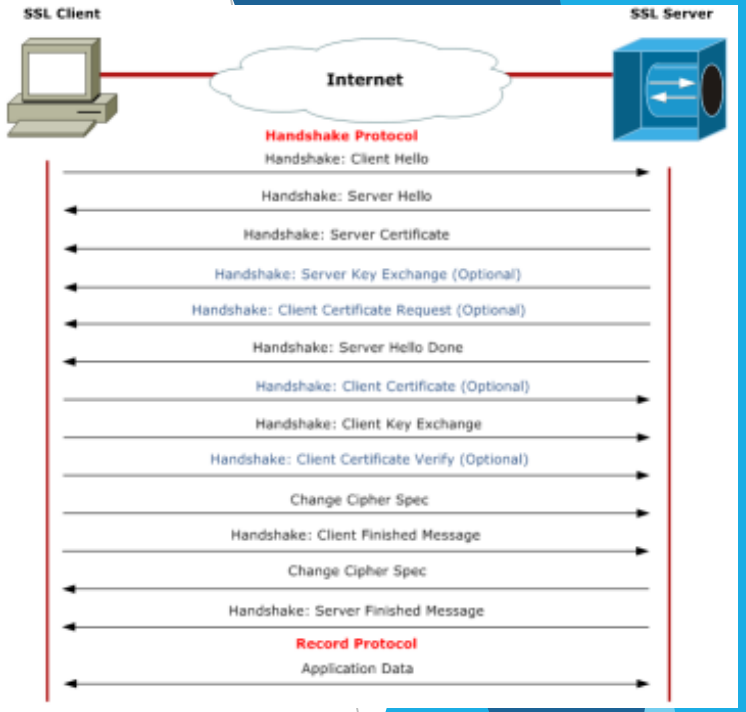
- Como a criptografia simétrica é mais eficiente para grandes volumes de dados, as chaves criptográficas são negociadas através do algoritmo Diffie-Hellman



Fonte: http://internetokracy.appspot.com/

Túneis e protocolos de segurança

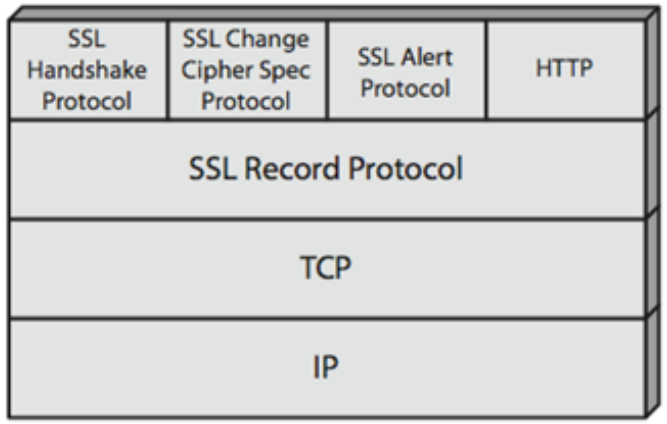
- *Secure Socket Layer* (SSL)
  - Protocolo desenvolvido pela Netscape que proporciona confidencialidade e integridade
  - Tornou-se um standard (RFC6101) mas abandonado posteriormente (RFC7568)
  - Opera em dois níveis
    - *SSL Handshake Protocol* (**SSLHP**) que opera no nível 7 do modelo OSI
    - *SSL Record Protocol* (**SSLRP**) que opera acima do nível 4 do modelo OSI
  - O **SSLHP** autentica os intervenientes na comunicação e negoceia a cifra a utilizar, criando o ***Master Secret*** a ser usado pelo **SSLRP**
  - O túnel criado tem um período de vida associado ao tempo de comunicação e/ou ao volume de dados transferidos
  - Se é detetada uma potencial quebra de segurança, o **SSLHP** termina de imediato a comunicação e reinicia o túnel
  - Após o cliente iniciar a ligação são trocadas mensagens através do **SSLHP** para estabelecer o túnel



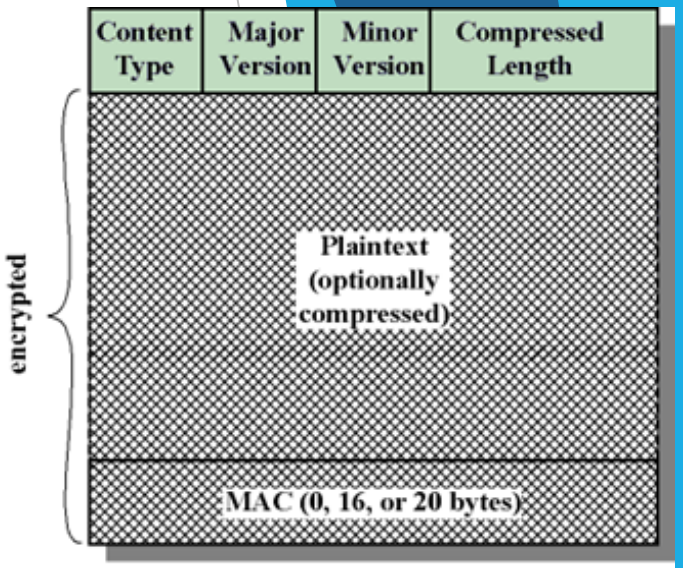
Fonte: Googlesites

Túneis e protocolos de segurança

- ***Secure Socket Layer*** **(SSL)**
  - Permite três modos de autenticação
    - Autenticação de ambas as partes
    - Autenticação do servidor sendo o cliente não autenticado
    - Modo anónimo
      - No modo anónimo é vulnerável a ataques ***Man-In-The-Middle***
  - O **SSLRP** captura as mensagens que vão ser trocadas, fragmenta-as em blocos com compressão opcional, aplicando a cada bloco a autenticação e depois a encriptação
  - No recetor a mensagem é desencriptada, verificada a sua integridade, expandido (se aplicável) e reagrupada

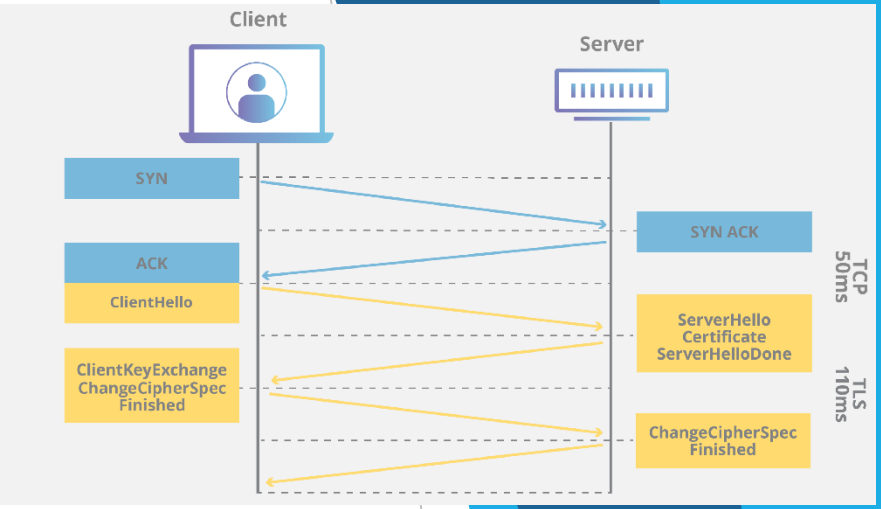


Fonte: StackExchange



Fonte: Mark Ryan, Birmingham University

Túneis e protocolos de segurança



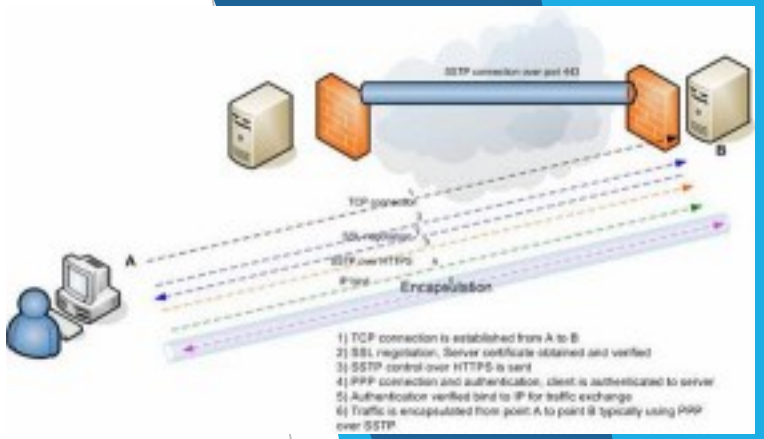
Fonte: Cloudflare

- ***Transport Layer Security*** (**TLS**)
  - Baseou-se no SSL não sendo contudo compatível com ele
  - Opera nos mesmos dois níveis, aqui designados ***TLS Handshake Protocol*** (**TLSHP**) e ***TLS Record Protocol*** (**TLSRP**), com as mesmas funções
  - A versão atual é a 1.3 (RFC 8446)
  - O modo de operação é igual à do SSL
    - O emissor captura a mensagem, fragmenta-a, autentica-a, comprime-a (opcional), encripta-a e transmite-a
    - O recetor desencripta, expande-a (se aplicável), verifica-a e reagrupa os blocos

Túneis e protocolos de segurança

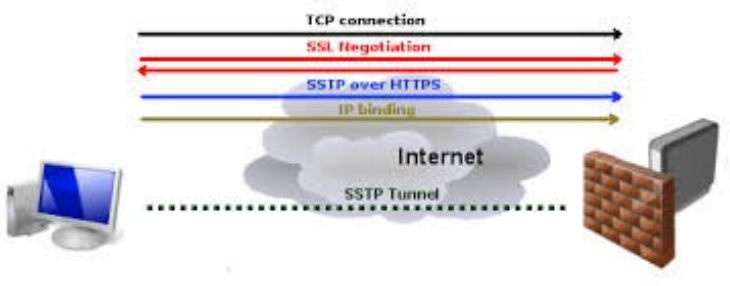
- ***Transport Layer Security*** **(TLS)**
  - Tem três modos de funcionamento
    - Sem autenticação
      - O túnel é estabelecido e negociada uma chave partilhada com o algoritmo de Diffie-Hellman
    - Autenticação do servidor
      - O servidor usa chaves assimétricas e um certificado para se autenticar
      - A chave pode ser negociada com o algoritmo de Diffie-Hellman ou escolhida pelo cliente que, neste caso, a envia ao servidor encriptando-a com a chave pública do servidor
    - Autenticação de ambos os sistemas
      - A chave é negociada de forma segura com recurso às chaves pública e privada de cada um
  - Os algoritmos criptográficos aplicáveis são à partida os mais recentes e seguros, sendo atualizados à medida que novos algoritmos são disponibilizados

Túneis e protocolos de segurança



Fonte: TechGenix

- ***Secure Socket Tunneling Protocol*** **(SSTP)**
  - Protocolo desenvolvido pela Microsoft, simples de implementar
  - Cria um túnel SSL/TLS no qual envia os dados
  - Os dados são encapsulados com recurso ao PPTP ou L2TP
  - A proteção que oferece é considerada boa, contudo o *overhead* inerente aos protocolos e algoritmos usados implica um menor desempenho, que pode ser compensado com um CPU potente e maior largura de banda
  - Alguns problemas foram identificados e merecem reflexão
    - É suscetível ao ***TCP Meltdown***
    - Não foi ainda realizada uma análise criptográfica por ser código fechado
    - Não suporta o uso de *proxy web*



Fonte: Sysinfo.io