

# Aula 20 - Segurança: WLANs, Firewalls, Gateways, IDS

Diego Passos

Universidade Federal Fluminense

Redes de Computadores II

# Na Última Aula...

- IPSec: provê segurança na camada de rede.
  - Confidencialidade, integridade, autenticação da origem.
- IPSec: dois protocolos.
  - AH: integridade e autenticação.
  - ESP: integridade, autenticação e confidencialidade.
- IPSec: dois modos.
  - Transporte: carga útil do datagrama IP é cifrada/autenticada.
  - Túnel: datagrama IP **completo é encapsulado** em novo datagrama.
    - Esconde **completamente protocolo de transporte, portas, ...**
- Associações seguras: SA.
  - Canal de comunicação virtual entre duas entidades IPSec.
  - **Simplex, mantém estado.**
    - Algoritmos de criptografia, integridade, chaves, ...
- IPSec: Gerenciamento de Chaves.
  - Protocolo próprio: IKE.

# Segurança em LANs Sem Fio

# Objetivos de Projeto do WEP

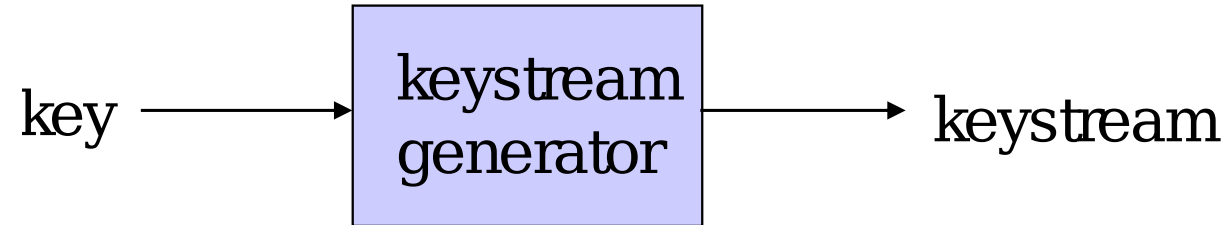
- Criptografia de chave simétrica.

- Confidencialidade.
- Autorização de hosts.
- Integridade dos dados.



- Auto-sincronização: cada pacote é cifrado separadamente.
- Dado pacote cifrado e chave, é possível decifrá-lo, mesmo que pacotes anteriores tenham sido perdidos (diferentemente de algoritmos como o *Cipher Block Chaining* (CBC) em cifras de bloco).
- Eficiência.
  - Implementável em *hardware* ou *software*.

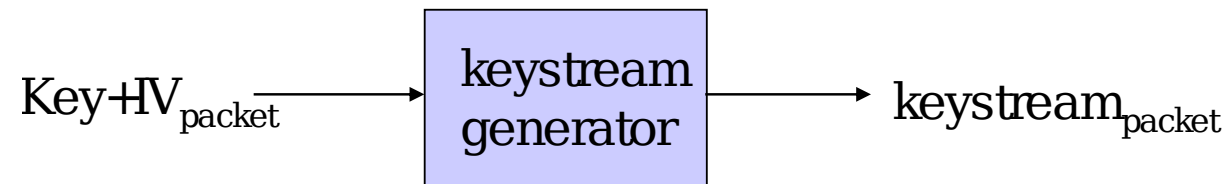
# Cifras de Fluxo Simétricas



- **Combina cada byte do fluxo de chave com byte de texto plano para obter texto criptografado:**
  - $m(i)$  =  $i$ -ésima unidade da mensagem.
  - $ks(i)$  =  $i$ -ésima unidade do fluxo de chave.
  - $c(i)$  =  $i$ -ésima unidade de texto criptografado.
  - $c(i) = ks(i) \oplus m(i)$
  - $m(i) = ks(i) \oplus c(i)$
- WEP usa o RC4.

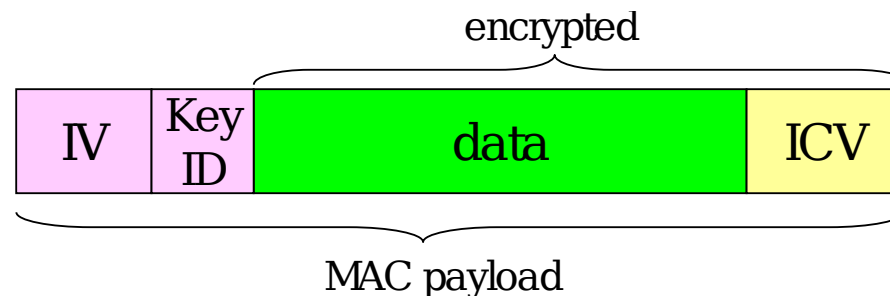
# Cifras de Fluxo e Independência de Pacotes

- Lembre-se de um objetivo de projeto: cada pacote é criptografado separadamente.
- Se para o quadro  $n+1$ , usamos o fluxo de chaves do ponto em que paramos no quadro  $n$ , então quadros não são criptografados separadamente.
  - É preciso saber onde paramos no último pacote.
- Abordagem do WEP: inicializa o fluxo de chave com a chave + um IV novo para cada pacote.
  - IV usado para cifrar um pacote é anexado **em texto plano** no próprio pacote.



# Criptografia no WEP (I)

- Transmissor calcula o *Integrity Check Value* (ICV) sobre os dados.
  - Hash/CRC de quatro bytes para verificação de integridade.
- Cada lado possui uma chave compartilhada de 104 bits.
- Transmissor cria um vetor de inicialização (IV) de 24 bits, adicionado à chave: resulta em nova chave de 128 bits.
- Transmissor também adiciona um keyID (em um campo de 8 bits).
- Chave de 128 bits é passada como entrada de um gerador de números pseudo-aleatórios para gerar o fluxo de chave.
- Dados no quadro + ICV são cifrados com o RC4:
  - Bytes do fluxo de chave são combinados através de um XOR com bytes dos dados e ICV.
  - IV e keyID são adicionados ao início dos dados criptografados para criar o *payload*.
  - *Payload* é encapsulado em um quadro 802.11.



# Criptografia no WEP (II)

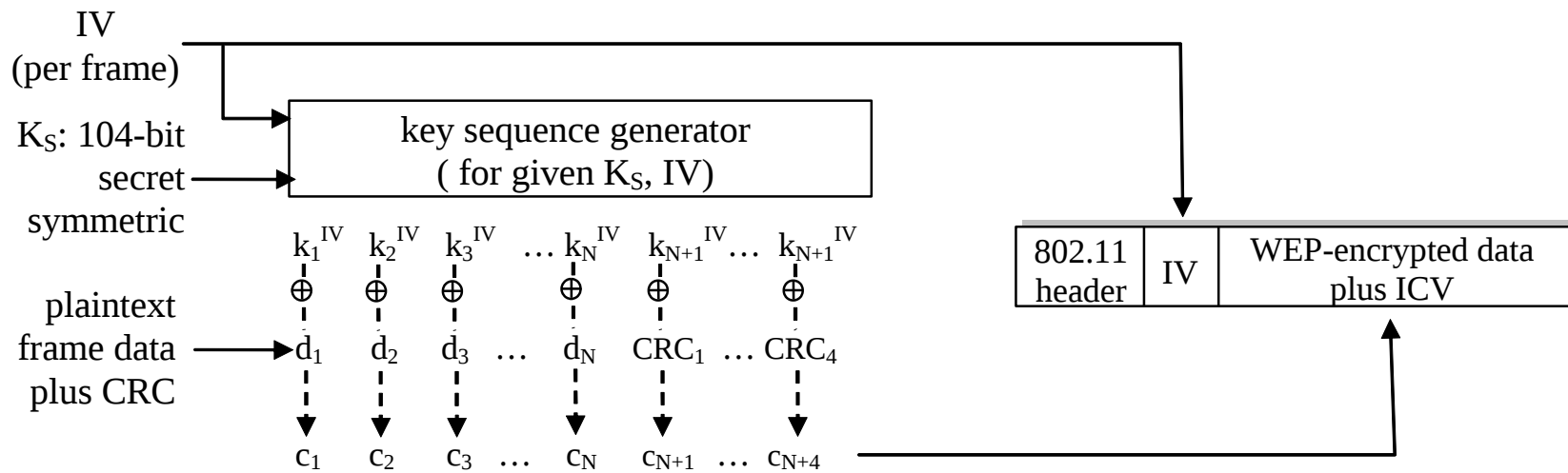
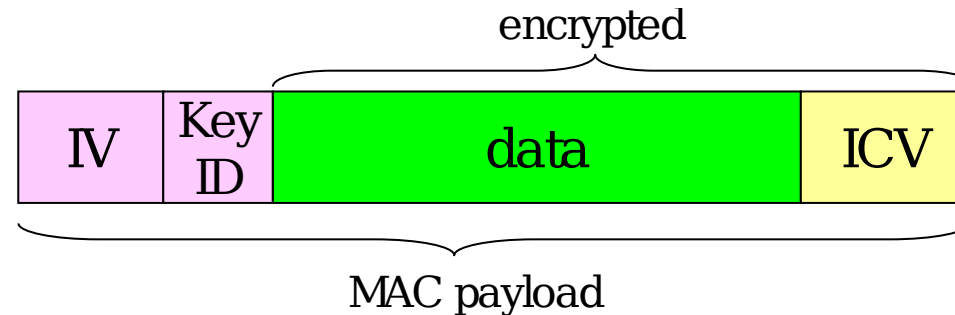


Figure 7.8-new1: 802.11 WEP protocol

**Um novo IV para cada quadro.**



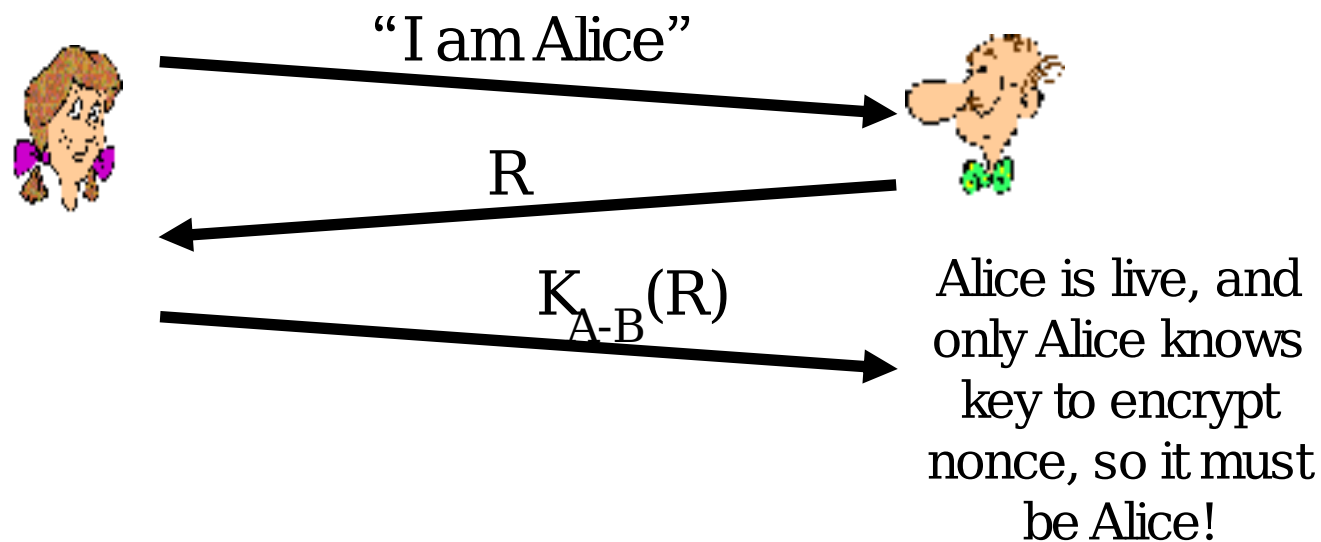
# WEP: Visão Geral do Processo de Deciframento



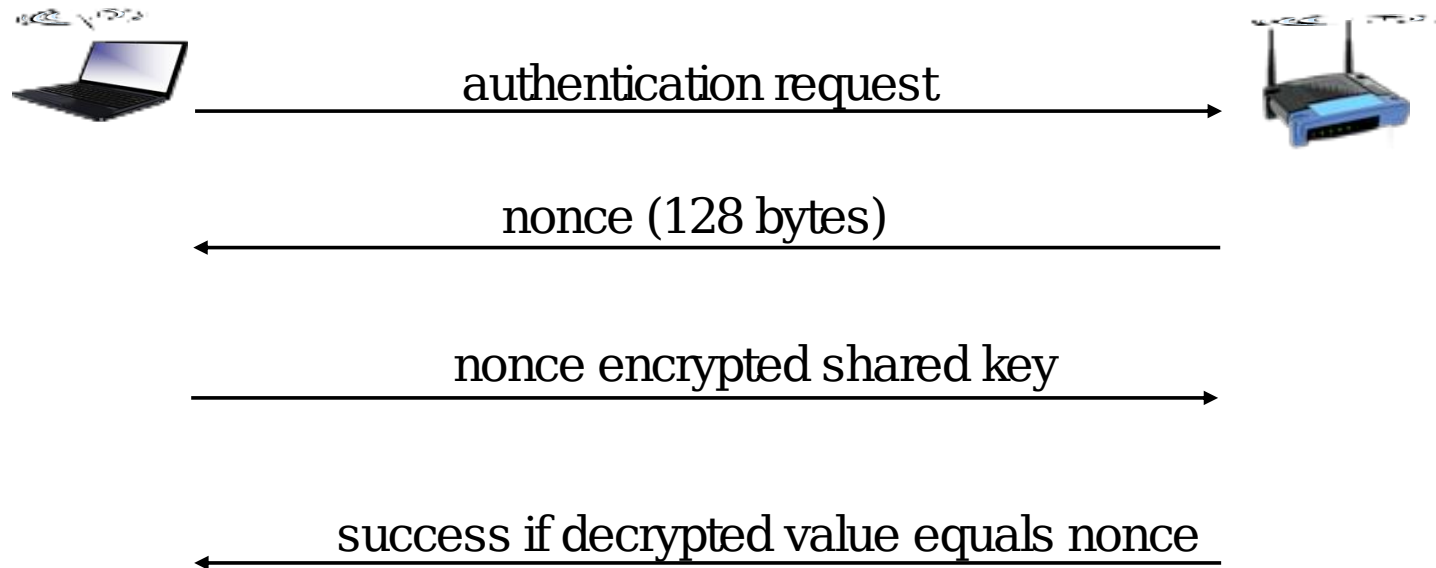
- Receptor extrai o IV.
- Passa IV e chave compartilhada como entrada do gerador de números pseudo-aleatórios, obtém fluxo de chave.
- Faz XOR entre fluxo de chave e dados criptografados para decifrar dados + ICV.
- Verifica integridade dos dados com o ICV.
  - Nota: abordagem de verificação de integridade usada aqui é diferente do MAC (Message Authentication Code) e assinaturas (usando PKI).

# Autenticação do Host Usando Nonce

- **Nonce:** número (R) usado “uma única vez”.
- **Como Alice prova estar “ao-vivo”:** Bob envia um **nonce**, R. Alice deve retornar R cifrado com a chave compartilhada.



# Autenticação WEP



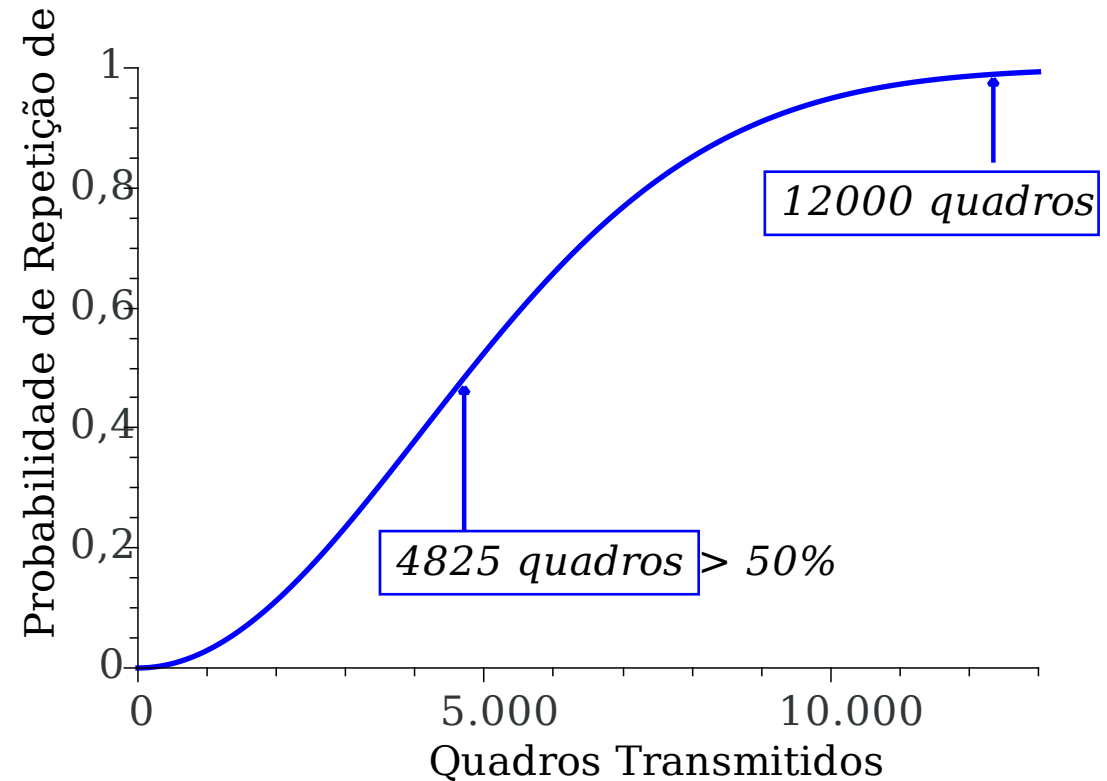
## ● Notas:

- Nem todos os APs realizam esta autenticação, mesmo quando WEP está em uso.
- AP indica se autenticação é necessária no quadro de *beacon*.
- Feito antes da associação.

# Quebrando a Criptografia WEP do 802.11

- **Problema de segurança:**

- IV de 24 bits, um IV por quadro  $\Rightarrow$  IVs eventualmente reutilizados.
- IV é transmitido em texto plano  $\Rightarrow$  reuso do IV é facilmente detectado.



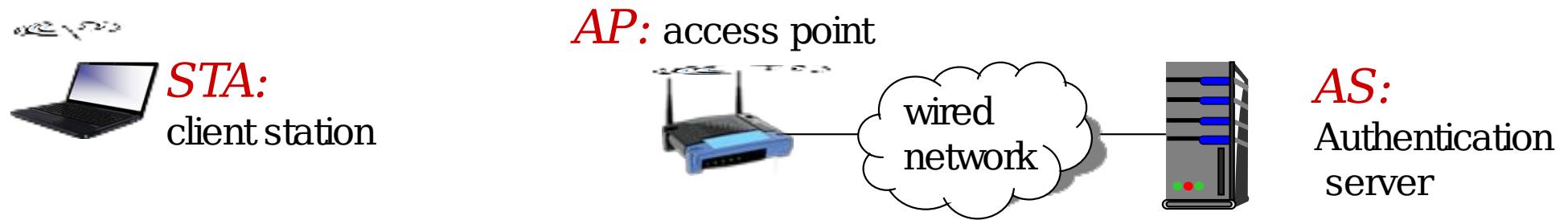
- **Exemplo de ataque:**

- Trudy força Alice a cifrar texto plano  $d_1d_2d_3d_4\dots$
- Trudy vê:  $c_i = d_i \oplus k_i^{IV}$ .
- Trudy conhece sequência de chaves  $k_1^{IV} \ k_2^{IV} \ k_3^{IV} \ k_4^{IV} \dots$  correspondente ao IV atual.
- Na próxima utilização do IV (para um quadro legítimo), Trudy pode decifrar!

## 802.11i: Melhorias de Segurança

- Várias formas (mais fortes) de criptografia são possíveis.
  - AES, ao invés de RC4.
- Provê sistema de distribuição de chaves.
  - “WPA Enterprise”, “WPA Personal”.
- No WPA Enterprise, usa servidor de autenticação separado do AP.
  - Dissociação credenciais de autenticação das chaves de criptografia.
  - Permite que cada usuário tenha suas próprias credenciais.

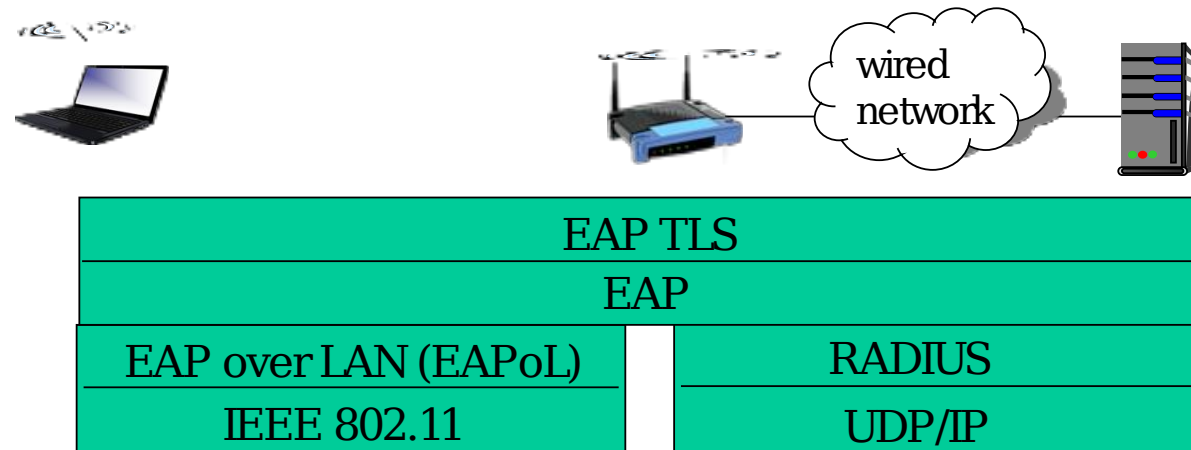
# 802.11i: Quatro Fases de Operação



- 1 Discovery of security capabilities
- 2 STA and AS mutually authenticate, together generate Master Key (MK). *AP serves as "pass through"*
- 3 STA derives Pairwise Master Key (PMK)
- 3 AS derives same PMK, sends to AP
- 4 STA, AP use PMK to derive Temporal Key (TK) used for message encryption, integrity

# EPA: Extensible Authentication Protocol

- EAP: protocolo de comunicação entre cliente (móvel) e servidor de autenticação.
- EPA enviado sobre enlaces “separados”.
  - Cliente móvel para AP (EAP sobre LAN).
  - AP para servidor de autenticação (Radius sobre UDP).

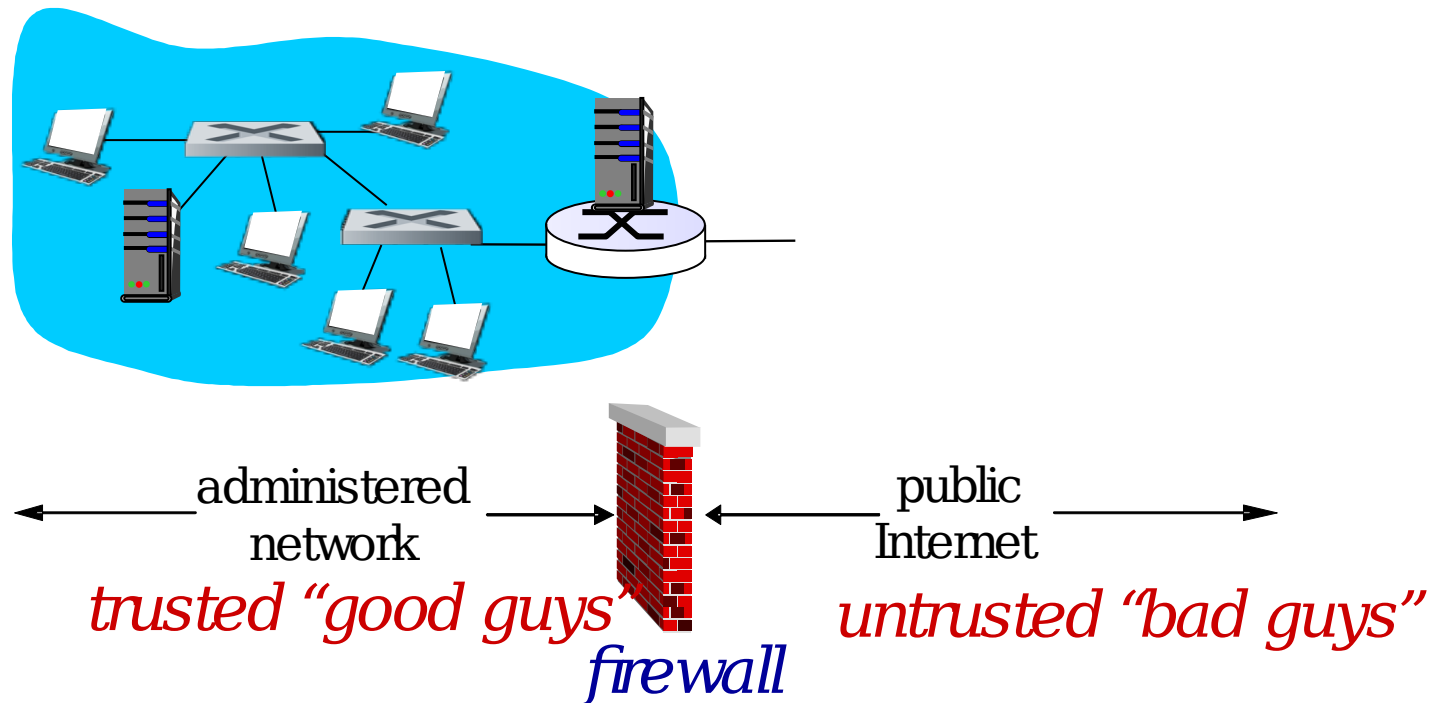


# Segurança operacional: Firewalls e IDS



# Firewalls

Isolam rede interna de uma organização da Internet pública, permitindo que alguns pacotes passem e bloqueando outros.



# Firewalls: Por Que?

- **Prevenir ataques de negação de serviço:**

- Inundação de SYNs: atacante estabelece várias conexões TCP falsas, não deixando recursos para conexões legítimas.

- **Prevenir modificações/acesso ilegais a dados internos:**

- e.g., atacante substitui página do governo por outro conteúdo.

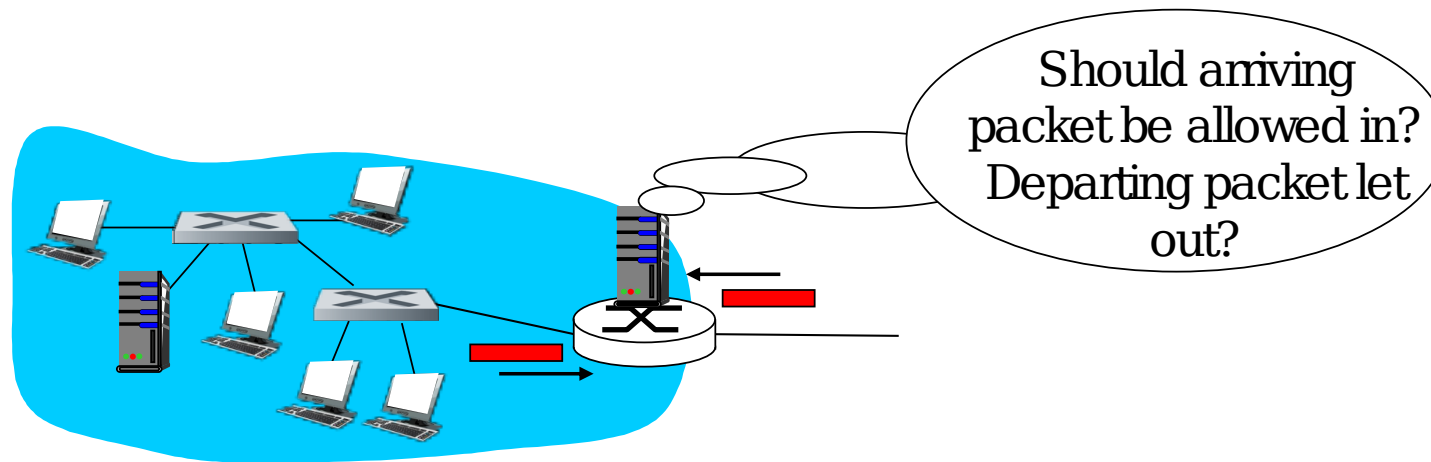
- **Permitir apenas acesso autorizado à rede interna:**

- Conjunto de usuários/hosts autenticados.

- **Três tipos de firewall:**

- Filtro de pacotes *stateless*.
- Filtro de pacotes *stateful*.
- Gateways de aplicação.

# Filtro de Pacotes Stateless



- Rede interna conectada à Internet através de um **roteador firewall**.
- Roteador **filtra pacote por pacote**, decidindo encaminhar ou descartar com base em:
  - Endereços IP de origem ou destino.
  - Números de porta de origem ou destino.
  - Tipo de protocolo de transporte (tipicamente, TCP ou UDP).
  - Tipo de mensagem ICMP.
  - Bits SYN e ACK em segmentos TCP.

# Filtro de Pacote Stateless: Exemplo

- **Exemplo 1:** bloqueie entrada e saída de pacotes com campo *protocol* do cabeçalho IP = 17. Bloqueie também pacotes com porta de origem ou destino = 23.
  - **Resultado:** todos os fluxos UDP entrando ou saindo e todas as conexões telnet são bloqueadas.
- **Exemplo 2:** bloqueie a entrada de segmentos TCP com bit ACK = 0.
  - **Resultado:** não permite que clientes externos abram conexões TCP com hosts internos, mas permite que hosts internos abram conexões para fora da rede.

# Filtro de Pacote Stateless: Mais Exemplos

| Política   | Configuração de Firewall  |
|--|---|
| Bloquear acesso Web.   | Descartar todos os pacotes para qualquer endereço IP, porta 80.                                       |
| Bloquear abertura de conexões TCP de fora para dentro, exceto aquelas direcionadas ao servidor Web da instituição. | Descartar todos os pacotes TCP com bit SYN ativo para qualquer IP, exceto 103.207.244.203, porta 80.  |
| Evitar que rádios on-line consumam toda a banda disponível.  | Bloquear a entrada de todos os pacotes UDP, exceto por pacotes de DNS e informações de roteamento.    |
| Evitar que sua rede seja usada para um ataque de DoS do tipo “smurf”.  | Descartar todos os pacotes ICMP com destino sendo um endereço de “broadcast” (e.g., 103.207.255.255). |
| Evitar que computadores externos façam um traceroute da sua rede.  | Descartar todas as mensagens ICMP de TTL expirado saindo da sua rede.                                 |

# Listas de Controle de Acesso

- **Access Control List (ACL):** tabela de regras, aplicadas de cima para baixo aos pacotes que chegam: pares do tipo (ação, condição).

| Ação     | Endereço de Origem | Endereço de Destino | Protocolo | Porta de Origem | Porta de Destino | Bit de Flag |
|----------|--------------------|---------------------|-----------|-----------------|------------------|-------------|
| Permitir | 222.22/16          | Fora de 222.22/16   | TCP       | > 1023          | 80               | Qualquer    |
| Permitir | Fora de 222.22/16  | 222.22/16           | TCP       | 80              | > 1023           | ACK         |
| Permitir | 222.22/16          | Fora de 222.22/16   | UDP       | > 1023          | 53               | ---         |
| Permitir | Fora de 222.22/16  | 222.22/16           | UDP       | 53              | > 1023           | ---         |
| Bloquear | *                  | *                   | *         | *               | *                | *           |

# Filtro de Pacotes Stateful (I)

- **Filtro de Pacotes Stateless:** ferramenta agressiva.
  - Admite pacotes que “não fazem sentido”, e.g., porta de destino 80, bit ACK ativo, embora não haja conexão TCP estabelecida:

| Ação     | Endereço de Origem | Endereço de Destino | Protocolo | Porta de Origem | Porta de Destino | Bit de Flag |
|----------|--------------------|---------------------|-----------|-----------------|------------------|-------------|
| Permitir | Fora de 222.22/16  | 222.22/16           | TCP       | 80              | > 1023           | ACK         |

- **Filtro de Pacotes Stateful:** monitora o estado de cada conexão TCP.
  - Monitora abertura (SYN) e fechamento (FIN): determina se pacotes que entram ou saem “fazem sentido”.
  - Conexões inativas sofrem *timeout* no firewall: pacotes não são mais admitidos.

# Filtro de Pacotes Stateful (II)

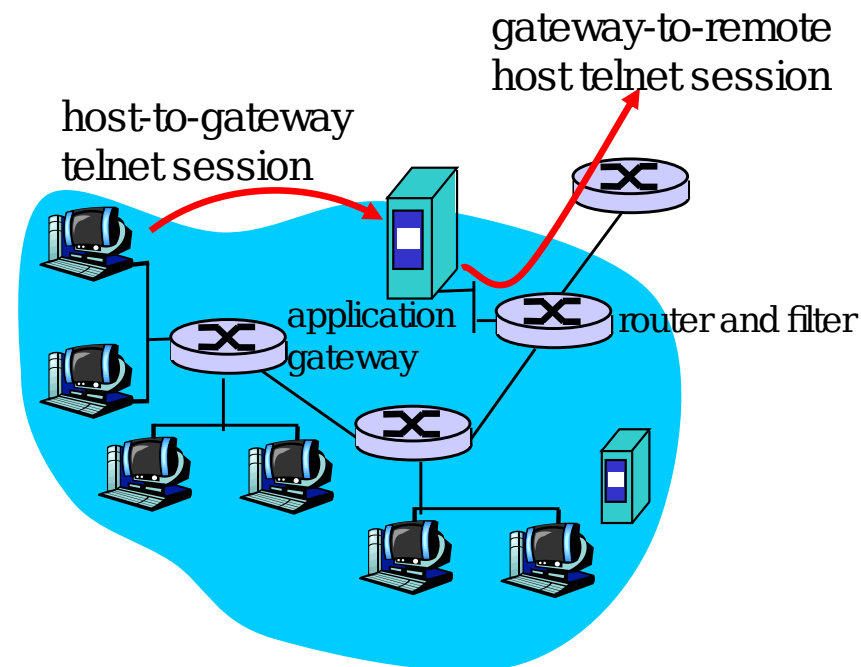
- ACL é aumentada para indicar necessidade de verificar tabela do estado das conexões antes de admitir um pacote.

| Ação     | Endereço de Origem | Endereço de Destino | Protocolo | Porta de Origem | Porta de Destino | Bit de Flag | Checar Conexão |
|----------|--------------------|---------------------|-----------|-----------------|------------------|-------------|----------------|
| Permitir | 222.22/16          | Fora de 222.22/16   | TCP       | > 1023          | 80               | Qualquer    |                |
| Permitir | Fora de 222.22/16  | 222.22/16           | TCP       | 80              | > 1023           | ACK         | X              |
| Permitir | 222.22/16          | Fora de 222.22/16   | UDP       | > 1023          | 53               | ---         |                |
| Permitir | Fora de 222.22/16  | 222.22/16           | UDP       | 53              | > 1023           | ---         | X              |
| Bloquear | *                  | *                   | *         | *               | *                | *           |                |



# Gateways de Aplicação (I)

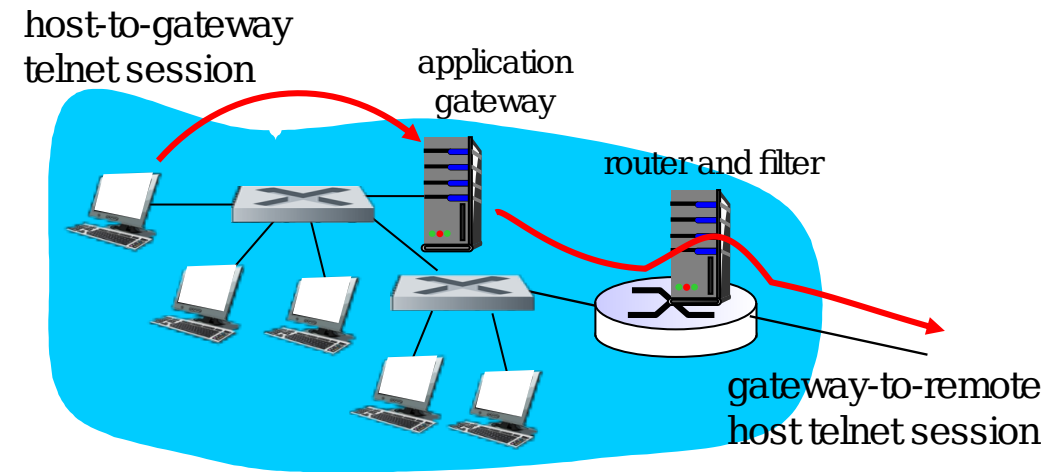
- Filtram pacotes baseados em dados da aplicação, além dos campos do IP/TCP/UDP.
- Exemplo: permite apenas que um subconjunto de usuários da rede realizem telnet para redes externas.



1. Requer que todos os usuários realizem telnet através do *gateway*.
2. Para usuários autorizados, *gateway* estabelece conexão com o host de destino. *Gateway* age como *relay* dos dados entre as duas conexões.
3. Filtro do roteador bloqueia todas as conexões telnet não originadas no *gateway*.

# Gateways de Aplicação (II)

- Filtram pacotes baseados em dados da aplicação, além dos campos do IP/TCP/UDP.
- Exemplo: permite apenas que um subconjunto de usuários da rede realizem telnet para redes externas.



1. Requer que todos os usuários realizem telnet através do *gateway*.
2. Para usuários autorizados, *gateway* estabelece conexão com o host de destino. *Gateway* age como *relay* dos dados entre as duas conexões.
3. Filtro do roteador bloqueia todas as conexões telnet não originadas no *gateway*.

# Limitações de Firewalls, Gateways

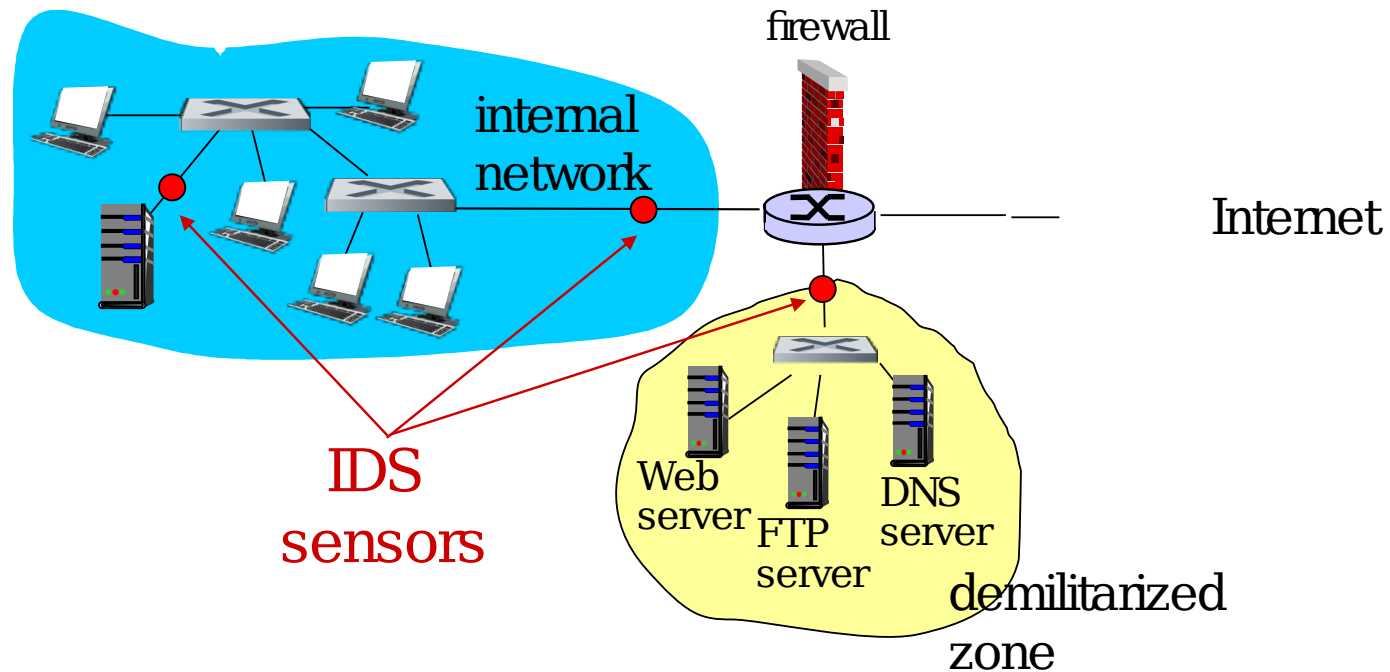
- **IP spoofing:** roteador não sabe se dados “realmente” vêm da origem identificada no cabeçalho IP.
- Se múltiplas aplicações precisam de tratamento especial, cada uma necessita do seu próprio *gateway*.
- *Software* do cliente deve saber como contactar o *gateway*.
  - e.g., através da configuração de um *proxy* no browser.
- Filtros tipicamente usam uma política do tipo “tudo ou nada” para UDP.
- **Compromisso:** grau de comunicação com o mundo externo, nível de segurança.
- Vários sites altamente protegidos ainda sofrem ataques.

# Sistemas de Detecção de Intrusão (I)

- Filtros de pacote:
  - Operam apenas sobre cabeçalhos TCP/IP.
  - Não verificam correlações entre sessões.
- **IDS: *Intrusion Detection System*.**
  - **Inspeção profunda nos pacotes:** olha o conteúdo do pacote (e.g., procura por padrões de bytes no pacote cadastrados em uma base de dados de vírus e ataques conhecidos).
  - **Verificação de correlação** entre vários pacotes.
    - *Port Scanning*.
    - Mapeamento da rede.
    - Ataque de DoS.

# Sistemas de Detecção de Intrusão (II)

- Vários IDSs: tipos diferentes de verificação em diferentes partes da rede.



# Resumo da Aula (I)...

- Segurança em WLANs: WEP.
  - **Padrão original** de segurança no IEEE 802.11.
  - Criptografia de **chave simétrica**.
  - Mesma chave era **compartilhada por todos os usuários**.
    - Chave usada tanto para confidencialidade, quanto para autenticação.
  - IVs para **evitar reuso** frequente de chaves.
    - IVs informadas em **texto plano** nos quadros.
    - 24 bits é pouco: **IVs se repetem rapidamente**.
  - Resultado: WEP é **extremamente vulnerável**.
- Segurança em WLANs: 802.11i.
  - Algoritmos **mais fortes**, chaves **maiores**, corrige vulnerabilidades conhecidas.
  - Versão “enterprise”: dissocia autenticação/confidencialidade, abole chaves compartilhadas.

# Resumo da Aula (II)...

- *Firewalls*: filtros de pacotes.
  - **Isolam** rede interna da Internet pública.
    - Aplicam **regras** para permitir/bloquear pacotes.
  - Previnem:
    - **Negação de serviço, acesso a dados internos, uso não autorizado, ...**
  - Podem ser.
    - *Stateless*: decisão baseada **apenas em campos do pacote** analisado.
    - *Stateful*: decisão leva em conta também **estado de conexões**.
    - *Gateways de Aplicação*: decisão baseada **dados de aplicação**.
- IDS: *Intrusion Detection System*.
  - Analisam tráfego, procuram por **padrões, assinaturas**.
  - Tentam detectar **comportamentos anômalos**.

# Leitura e Exercícios Sugeridos

- Segurança em WLANs:
  - Páginas 531 a 535 do Kurose (Seção 8.7).
  - Exercícios de fixação 27 e 28 do capítulo 8 do Kurose.
  - Problema 24 do capítulo 8 do Kurose.
- Segurança Operacional (*firewalls*, *gateways* de aplicação e IDS):
  - Páginas 535 a 544 do Kurose (Seção 8.8).
  - Exercícios de fixação 29, 30, 31 e 32 do capítulo 8 do Kurose.
  - Problema 25 do capítulo 8 do Kurose.
- Leitura adicional (opcional): anonimado e privacidade.
  - Página 541 do Kurose.
  - Problema 26 do capítulo 8 do Kurose.



# Segurança em Redes (Sumário)

- **Técnicas Básicas...**
  - Criptografia (simétrica e pública).
  - Integridade de mensagens.
  - Autenticação das partes.
- **...usadas em vários cenários de segurança:**
  - E-mail seguro.
  - Transporte seguro (SSL).
  - IPsec.
  - 802.11.
- **Segurança operacional: firewalls e IDS.**

# Próxima Aula...

- Começaremos um novo tópico: redes multimídia.
  - Aplicações típicas.
  - Requisitos.
  - Estudos de caso.
  - Protocolos.
  - Qualidade de serviço.