Tópicos Avançados em Computação I
(Criptografia)

Códigos de Autenticação de Mensagens

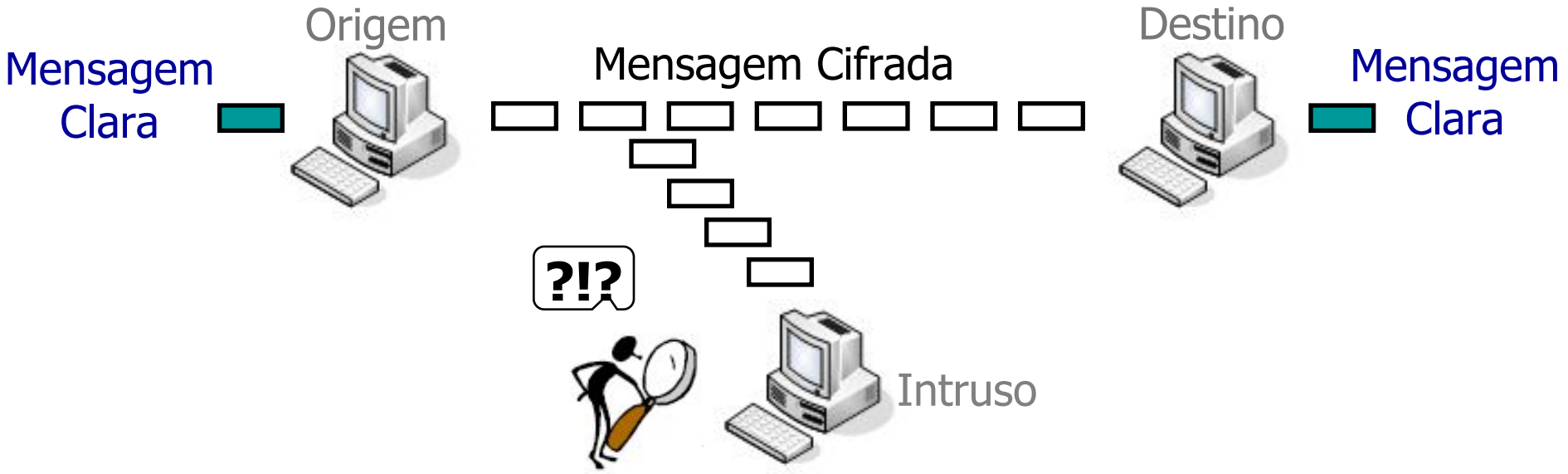
Prof. Dr. Ewerton R. Andrade – ewerton.andrade@unir.br

Observação: Aula adaptada do material produzido pelo Prof. Dr. Marcos A. Simplicio Jr. para a disciplina de Segurança em Redes de Computadores da Escola Politécnica da USP.

UNIR

Nos episódios anteriores…

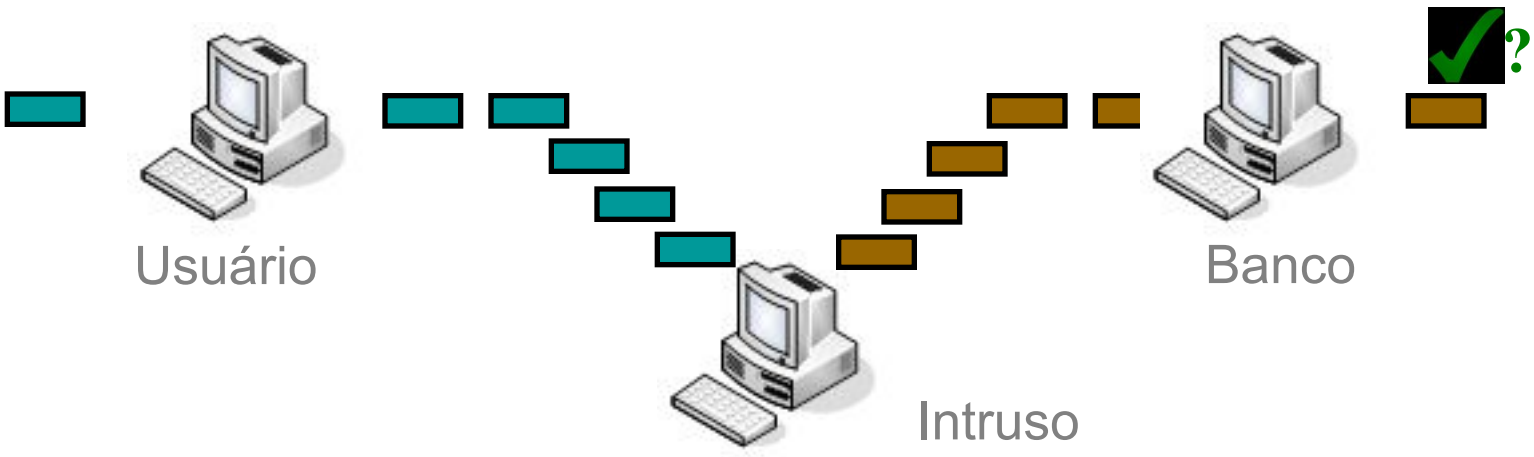
- Confidencialidade
  - Capacidade de prevenir vazamento de informações



"Seu saldo é R$10.000,00" → "Hlaafd7Y(@&fhF23%7" → "Seu saldo é R$10.000,00"

Nos episódios anteriores…

- Integridade
  - Capacidade de verificar se informação foi alterada



Integridade: redundância

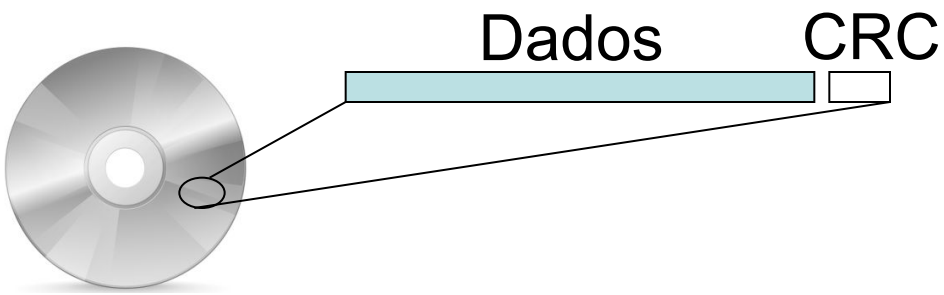
- Exemplo prático (não-criptográfico):
  - RG/CPF: usa Dígito verificador (DV)
  - Método: “mod 11”
    - Dígito é multiplicado por sua posição, indo do menos significativo (peso 2) até o mais significativo
    - Os resultados são somados
    - DV: resto da divisão desta soma por 11

Exemplo simplificado

Entrada:	**2**	**3**	**5**	**9**	**2**
Posição:	**6** x	**5** x	**4** x	**3** x	**2** x
Multiplicação:	**12**	**15**	**20**	**27**	**4**
Soma:	**78**				Σ
DV:	**78 mod 11 = 1**				mod 11

Integridade: redundância

- Exemplo prático (não-criptográfico):
  - CD/DVD: usa Cyclic Redundancy Check (CRC)

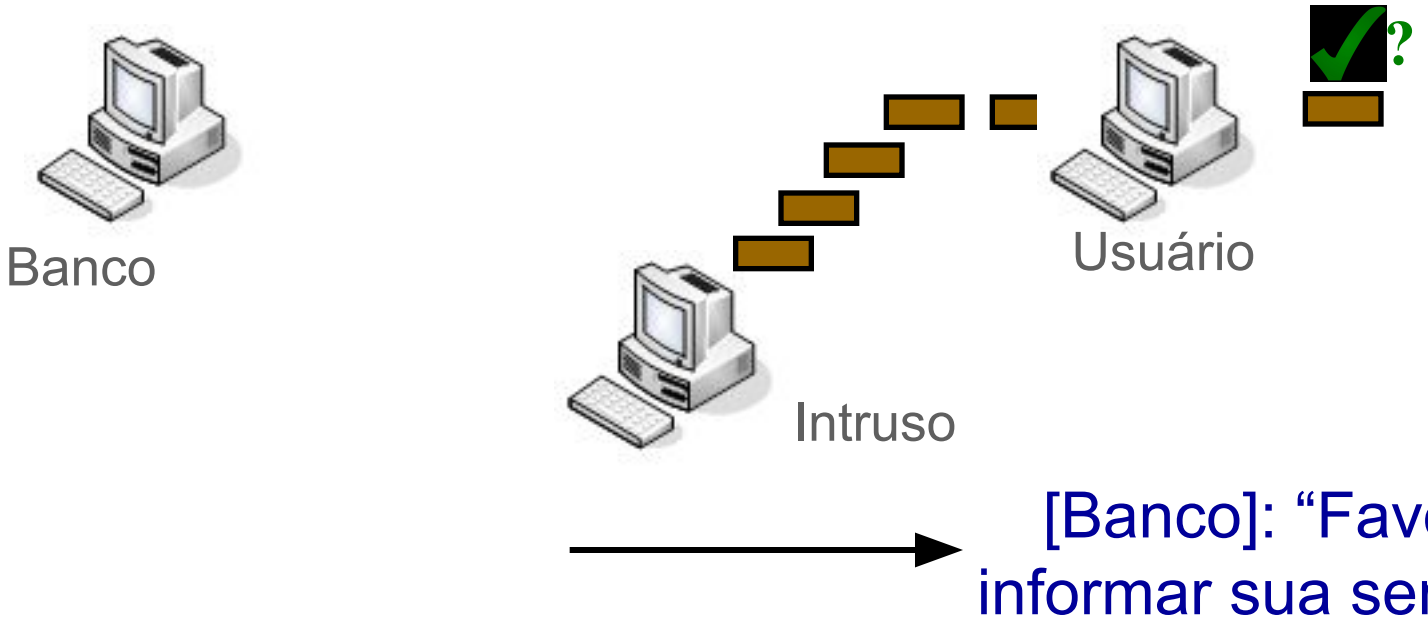


- Se Dados forem alterados (ex.: CD riscado)
  - FunçãoVerificação(Dados) ≠ CRC
    - Computador acusa erro de leitura!

Códigos de Autenticação

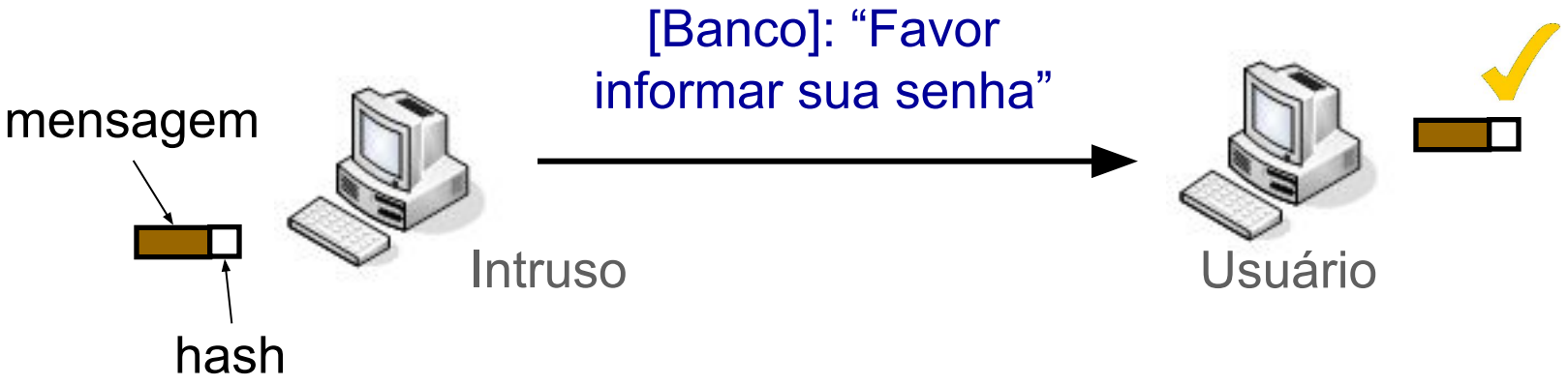
Nos episódios anteriores…

- Autenticidade
  - Capacidade do receptor em verificar quem é o emissor da mensagem



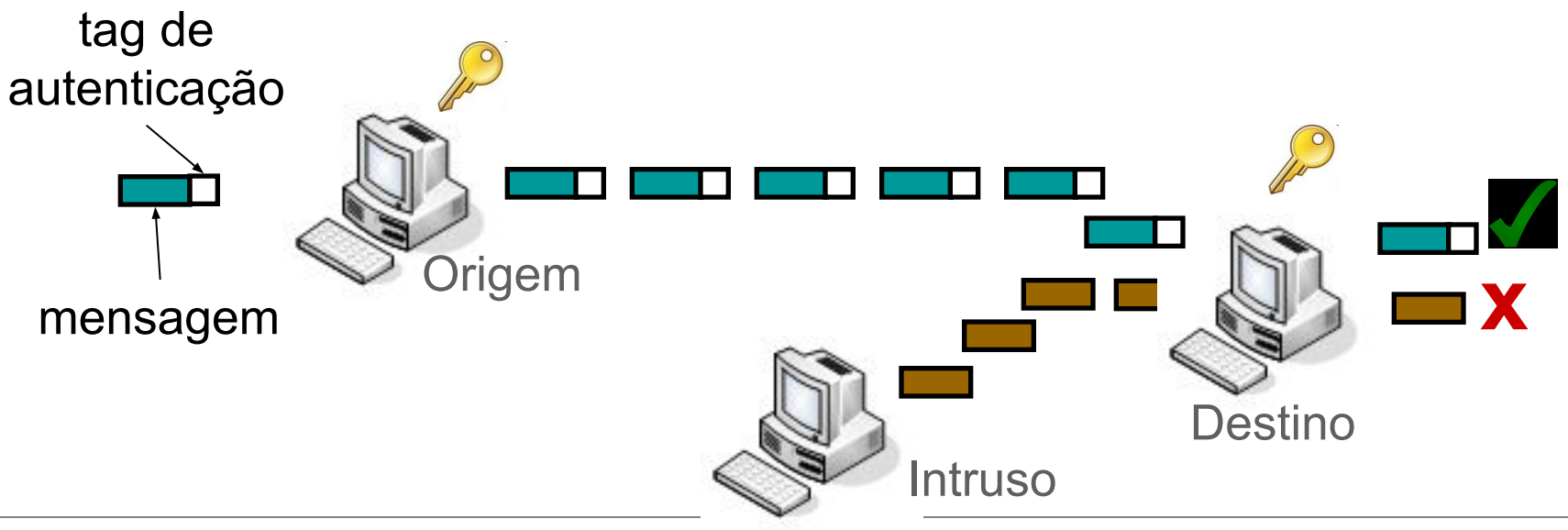
Usar hash?

- Hash sozinho não funciona…
  - Qualquer pessoa (incluindo intruso) pode calcular o hash da mensagem falsa…
  - O fato da mensagem estar íntegra não significa que foi o banco quem a enviou…



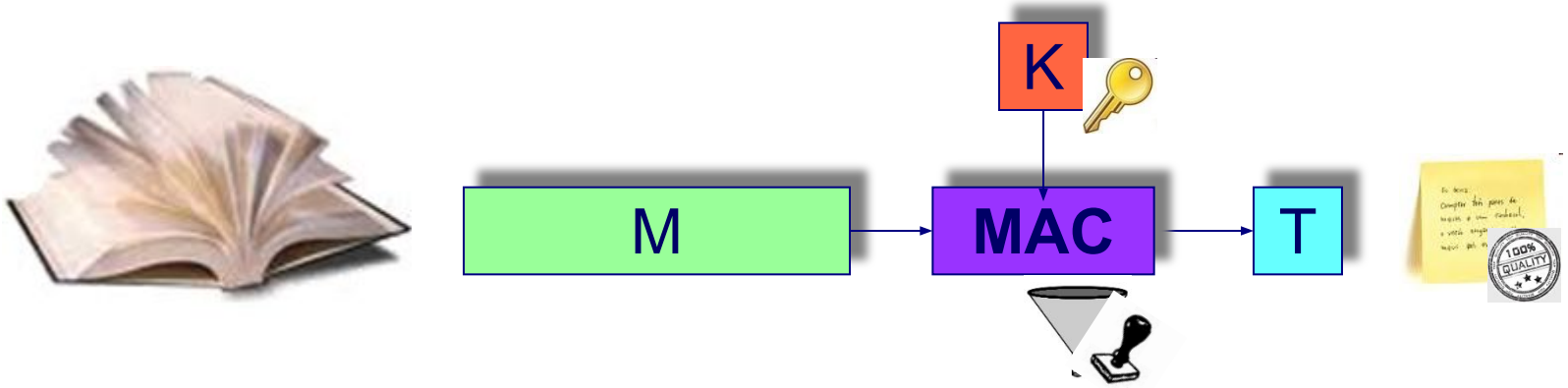
Estratégia

- Usar ***<u>redundância dependente de chave</u>***
  - Apenas origem e destino conhecem a chave e conseguem calcular redundância corretamente
  - Também garante integridade (alteração na mensagem detectada como no caso das funções de hash)

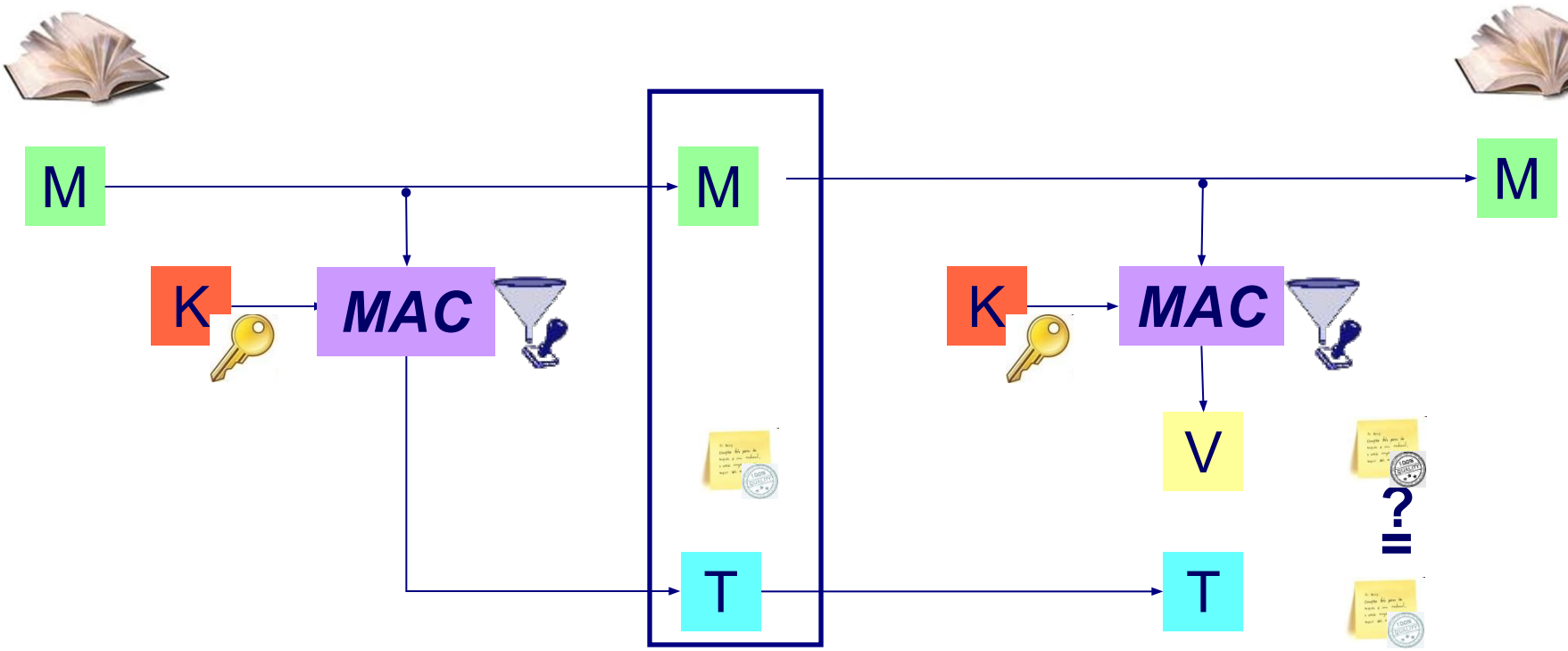


Códigos de Autenticação

- Redundâncias anexadas a mensagens de modo a detectar alterações (***integridade***) e garantir a ***autenticidade*** do remetente.
  - Chamada de “tag (etiqueta) de autenticação”
- Dependem da mensagem e também de uma informação secreta, compartilhada entre o remetente e o destinatário.
  - Propriedades de segurança: semelhantes a hash + incapacidade do atacante em recuperar a chave



Códigos de Autenticação: uso



- Envio de mensagem ***autenticada***
  - K: chave simétrica compartilhada
  - T: tag → garante integridade e autenticidade

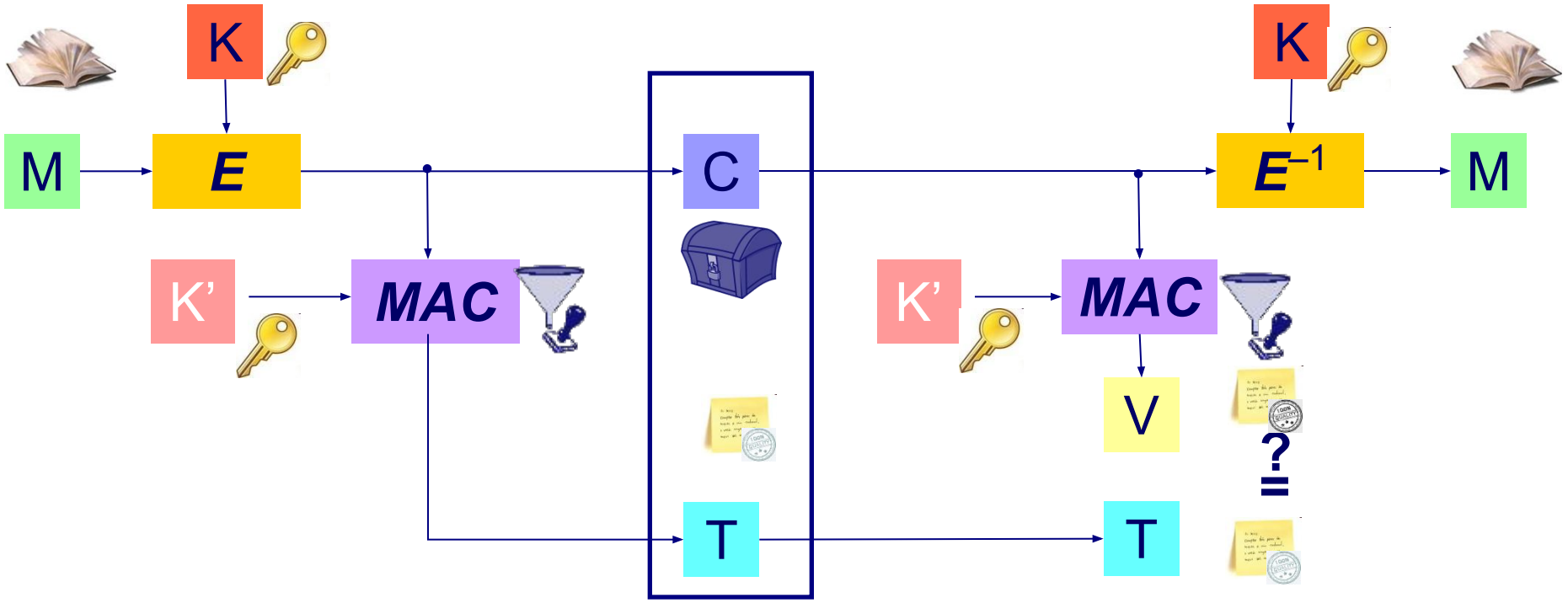
Construções Comuns

- Cifras de bloco:
  - CBCMAC (FIPS 113, ANSI X9.17).
  - CMAC (NIST SP 800-38B).
  - Vantagem: espaço de código (aproveitam implementações existentes de cifras de bloco).
- Funções de *hash*:
  - HMAC (FIPS 198).
  - Vantagem: velocidade de operação (funções de *hash* puras).
- As duas estratégias são bastante usadas na prática

Cuidados de Uso

- Uma mesma chave **não** deve ser utilizada para **cifrar e autenticar** mensagens
- Cada algoritmo tem suas próprias restrições de segurança
  - Número máximo de mensagens que podem ser autenticadas usando uma mesma chave
  - Tamanho máximo da mensagem autenticada
  - Uso apenas com mensagens de tamanho fixo (ex.: CBC-MAC) ou de qualquer tamanho (ex.: CMAC e HMAC)

MAC + cifra (uso no TLS)



- Mensagem ***confidencial*** (C) e ***autenticada*** (T)
  - K e K': chaves compartilhadas diferentes
  - Serviços: confidencialidade (cifra simétrica), integridade e autenticidade (algoritmo de MAC)

Assinaturas Digitais?

- Um código de autenticação pode garantir *Integridade* e *Autenticidade*.
- Não pode garantir *irretratabilidade*, pois tanto o remetente quanto o destinatário conhecem a mesma chave.
  - Não é possível provar para um **terceiro** quem de fato gerou o código de autenticação!
- Numa assinatura digital verdadeira, apenas o remetente conhece a chave de assinatura.