Serviços Distribuídos Tolerantes a Intrusões

uma introdução

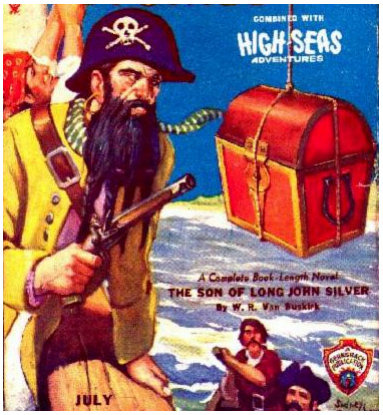
Miguel Correia
mpc@di.fc.ul.pt
www.di.fc.ul.pt/~mpc

Grupo Navigators, LASIGE
Faculdade de Ciências da Universidade de Lisboa

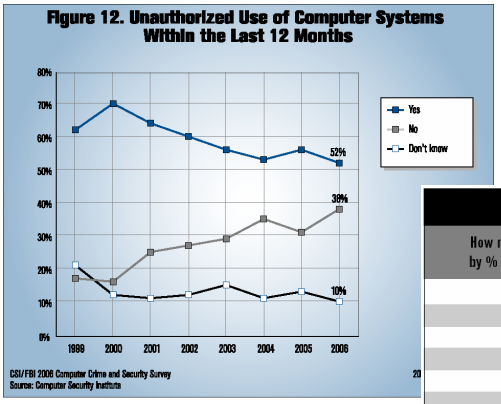
PUC-Paraná – Curitiba – 12 de Setembro de 2006

Uma estória

Final (in)feliz



Dados recentes de (in)segurança



CSI/FBI 2006 Computer Crime and Security Survey

Inquéritos a empresas, agências governamentais, instituições financeiras, universidades

Table 1: How Many Incidents?

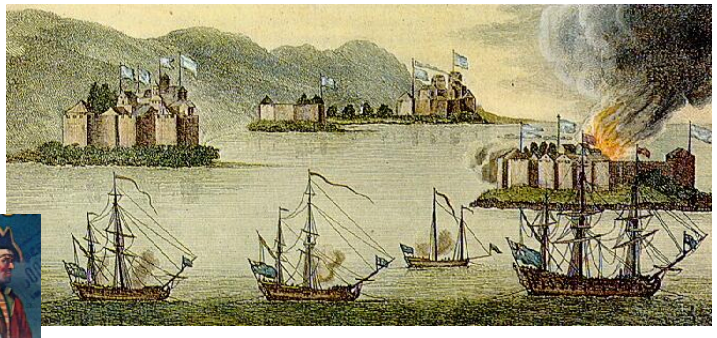
How many incidents, by % of respondents	1–5	6–10	>10	Don't know
2006	48	15	9	28
2005	43	19	9	28
2004	47	20	12	22
2003	38	20	16	26
2002	42	20	15	23
2001	33	24	11	31
2000	33	23	13	31
1999	34	22	14	29

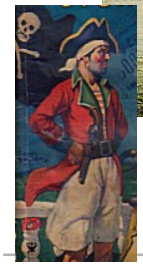
CSI/FBI 2006 Computer Crime and Security Survey
Source: Computer Security Institute
2006: 341 Respondents

Tolerância a Intrusões

- Segurança
  - ☞ênfase na prevenção: muralha, guardas, jacarés, ... – firewalls, controle de acesso, cripto...
- Confiança no Funcionamento (CnF)
  - ☞procura que o sistema continue operacional mesmo que alguns componentes falhem
  - ☞se o computador de bordo do avião falhar...
- Tolerância a Intrusões (TI)
  - ☞*aplicar o paradigma da tolerância a faltas no domínio de segurança*

Aplicando TI à estória





redundância
diversidade

A área da TI

- Conceito com 20 anos:
  - Joni Fraga e David Powell. A fault- and intrusion-tolerant file system. *Proc. Int'l Conf. on Computer Security*, 1985
- Por volta de 2000:
  - projecto europeu MAFTIA
  - programa americano OASIS (projectos ITUA, SITAR, PASIS, AgileStore...)
  - …

O palestra

- A TI é muito vasta
  - p.ex. podia-se dizer que a *detecção de intrusões* é um mecanismo de TI (mas a origem é diferente e tem mérito próprio!)
- Trabalho mais interessante dos últimos anos: **serviços distribuídos TI** (minha opinião claro...)
- *Objetivo:*
  garantir integridade, disponibilidade e confidencialidade de **serviços** constituídos por diversos servidores ligados por uma rede
  mesmo que alguns servidores sejam atacados e controlados por atacantes ou código nocivo

Serviços distribuídos TI

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

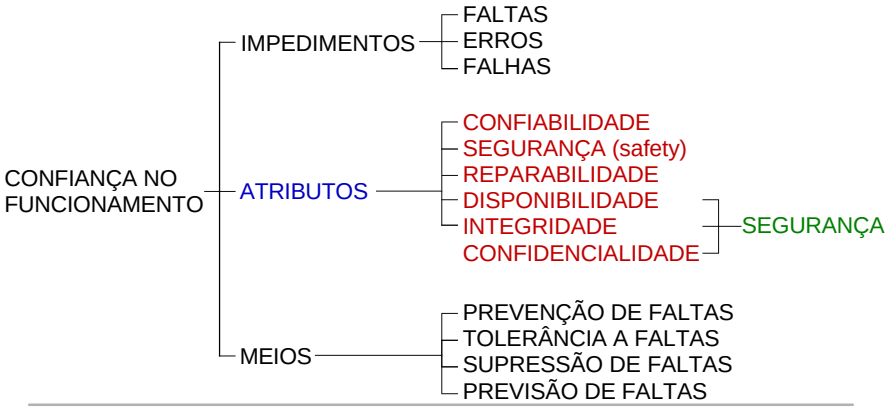
Confiança no Funcionamento

Processo de falha

- sistema oferece um serviço correto se obedece à sua especificação;
- caso contrário **falha** – o que queremos evitar!
- **falta**: causa remota de uma falha
  - ☞interna – p.ex. defeito numa RAM
  - ☞externa – p.ex. operador tropeça e desliga cabo
- **erro**: consequência de uma falta no estado
  - ☞p.ex. registo corrompido devido a defeito na RAM
- falta ? erro ? falha

Atributos de CnF

- Objetivo da CnF é garantir um conjunto de atributos de um sistema apesar da ocorrência de faltas:



Tolerância a Intrusões

Tolerância a Intrusões

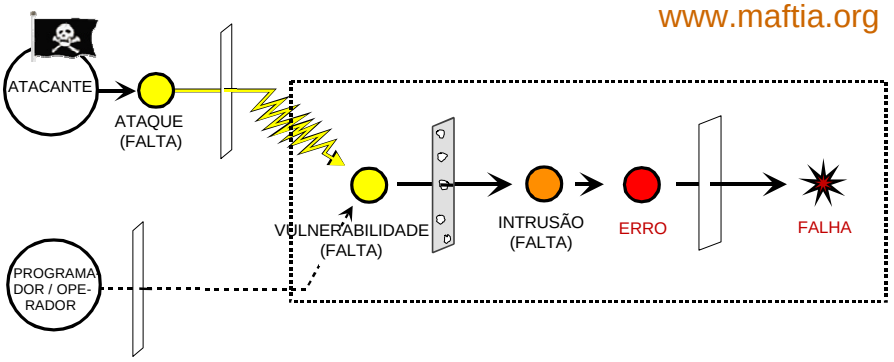
- O que é *aplicar o paradigma da tolerância a faltas no domínio de segurança?*
  - ☞assumir e aceitar que o sistema permanece sempre mais ou menos vulnerável;
  - ☞assumir e aceitar que os componentes do sistema podem ser atacados e que alguns desses ataques terão sucesso;
  - ☞garantir que o sistema como um todo permanece seguro e operacional, ou seja, que não falha.

Conceitos de TI

- **vulnerabilidade**: falta de projeto ou de configuração, geralmente acidental, que pode ser explorada com fins maliciosos
- **ataque**: falta intencional, maliciosa, que visa explorar uma ou mais vulnerabilidades
- **intrusão**: resultado de um ataque que tem sucesso em explorar uma ou mais vulnerabilidades
- Estas faltas são englobadas na categoria mais geral: faltas arbitrárias ou bizantinas
  - ☞ L. Lamport et al., The Byzantine Generals Problem, 1982

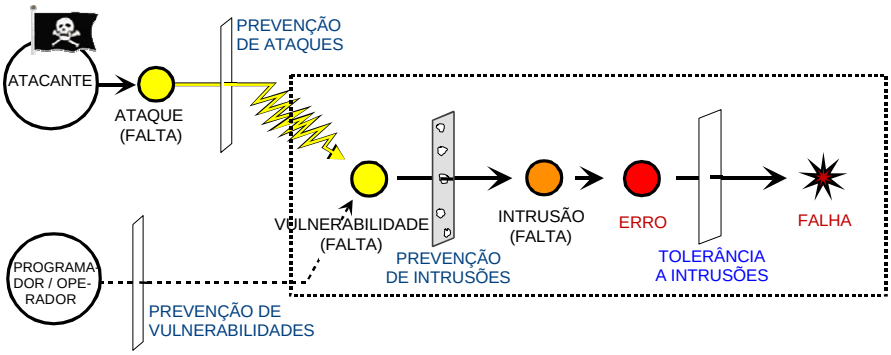
Modelo AVI

Projeto MAFTIA
www.maftia.org



processo de falha

Modelo AVI (cont)



processo de falha; meios para a evitar

Meios de CnF e serv.dist.TI

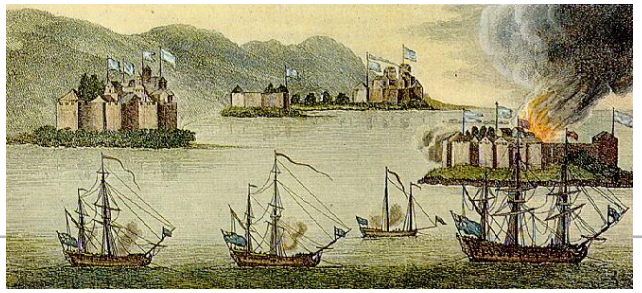
- O projecto de serviços distribuídos TI envolve os quatro meios de CnF:
- Tolerância a faltas:
  - ☞a maior parte das soluções que veremos usam mascaramento de faltas: redundância de máquinas + protocolos tolerantes a faltas bizantinas
  - ☞processamento de erros: recuperação proativa
- Prevenção, supressão e previsão de faltas são importantes mas não são específicos da TI
  - ☞TI não substitui os meios de Segurança!!

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

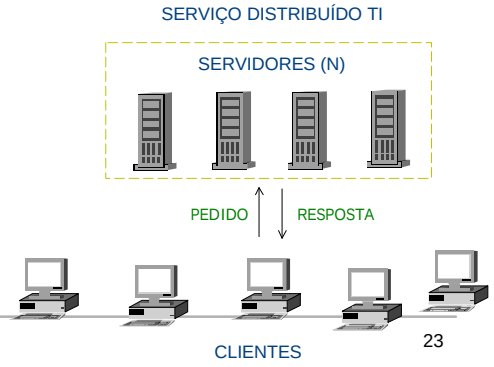
Replicação

- Idéia: código e dados replicados nos servidores
- objetivo: garantir a **disponibilidade** e a **integridade** do serviço e/ou dados (atributos)
- Duas aproximações: **replicação de máquinas de estados** e **sistemas de quoruns**



Replicação de Máq.de Estados

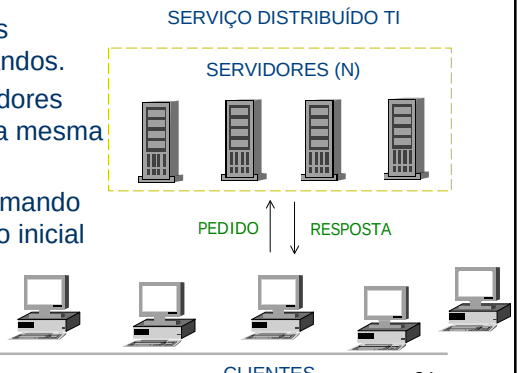
- (ou **replicação ativa**) – solução genérica para a concretização de *serviços* tolerantes a faltas
- cada servidor é uma máquina de estados definida por variáveis de estado; comandos atômicos



Replicação de Máq.de Estados

- todos os servidores seguem a mesma sequência de estados sse:
- Estado inicial. Todos os servidores começam no mesmo estado.
- Acordo. Todos os servidores executam os mesmos comandos.
- Ordem total. Todos os servidores executam os comandos pela mesma ordem.
- Determinismo. O mesmo comando executado no mesmo estado inicial gera o mesmo estado final.

protocolo de difusão atômica



Resistência

- número máximo **f** de **servidores** que podem falhar (ser corrompidos) para o serviço se manter correto
- em sistemas assíncronos TI baseados em replicação de máquinas de estado este limite é imposto pelo *protocolo de difusão atômica*: $N = 3f+1$
  - ex: N=4 servidores para tolerar f=1 corrompido; N=7 para tolerar f=2
- não há limite ao nº **clientes** que podem falhar
- cobertura das premissas – pode-se definir **f**?
- Sistemas: Rampart, BFT, SecureRing, SecureGroup, SINTRA, Worm-IT

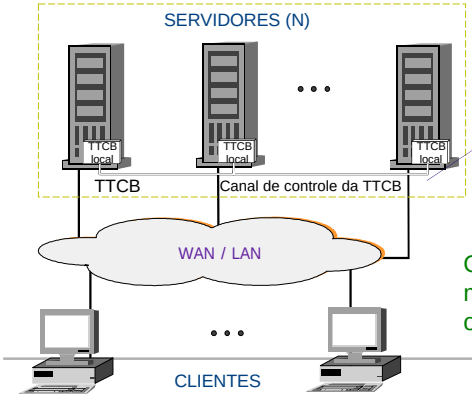
Resistência

- Será que a resistência é indiferente?
- Cada réplica tem três custos
  - hardware e software
  - projeto (código feito à medida...)
  - gerenciamento
- Diminuir o número de réplicas é importante!
- Mas em sistemas assíncronos já vimos que o mínimo é N = 3f+1 por causa do protocolo de difusão atômica

Resistência 2f+1

M. Correia, N. Neves,
P. Veríssimo, U. Lisboa

- Usando um **modelo de falhas híbrido arquitetural** consegue-se diminuir o nº de servidores de **3f+1** para **2f+1** (25% a 33%)



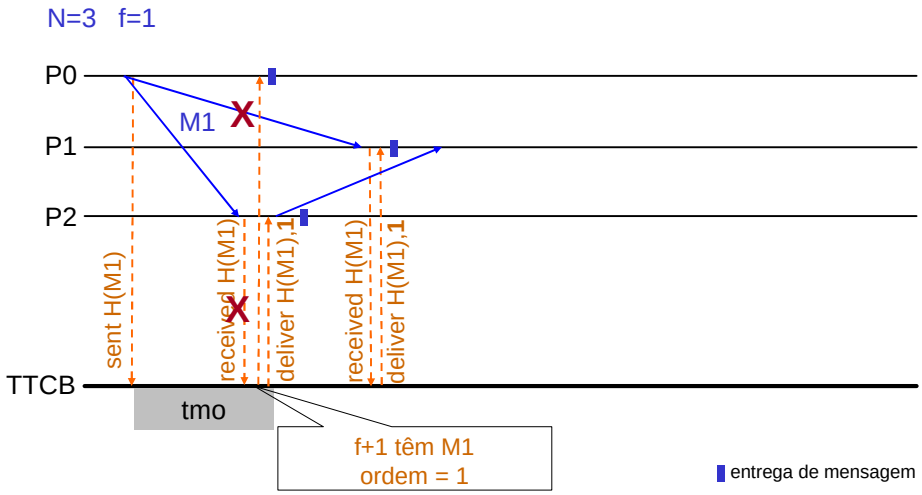
WORMHOLE: subsistema seguro e com funcionalidade limitada

O serviço é executado nos servidores mas chama a TTCB para ajudar a ordenar os pedidos.

Serviços da TTCB

- Trusted Multicast Ordering Service (TMO)
  - ☞O objetivo é suportar a execução de um protocolo de difusão atômica TI
  - ☞Não é afectado por faltas bizantinas (ataques, intrusões) pois é executado dentro da TTCB
- (existem outros mas não são necessários aqui)

Difusão atômica com o TMO



Serviço TMO

- Núcleo da solução
  - ☞Decide quando uma mensagem pode ser entregue
    - – se **f+1** servidores mostram que têm a mensagem, então pelo menos um correto tem
  - ☞Define uma ordem sequencial para as mensagens
  - ☞Resultados são confiáveis pois a TTCB é segura
- Concretização do serviço
  - ☞Quando há uma mensagem, TTCBs locais executam um *protocolo de acordo* para decidirem o nº de ordem
  - ☞Esse protocolo é executado num ambiente benigno, não tem de tolerar faltas bizantinas

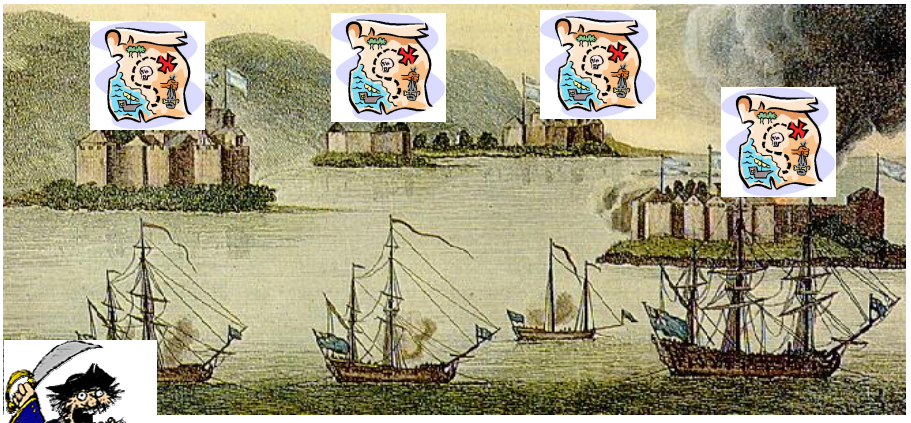
Envio de pedidos

- Os clientes têm relógios locais (não confiáveis) Protocolo:
  - Enviar o pedido para um servidor
  - Esperar por **f+1** respostas idênticas de servidores diferentes
  - Se T_{resend} depois do pedido ter sido enviado não tiverem sido recebidas as respostas, reenviar para **f** servidores adicionais

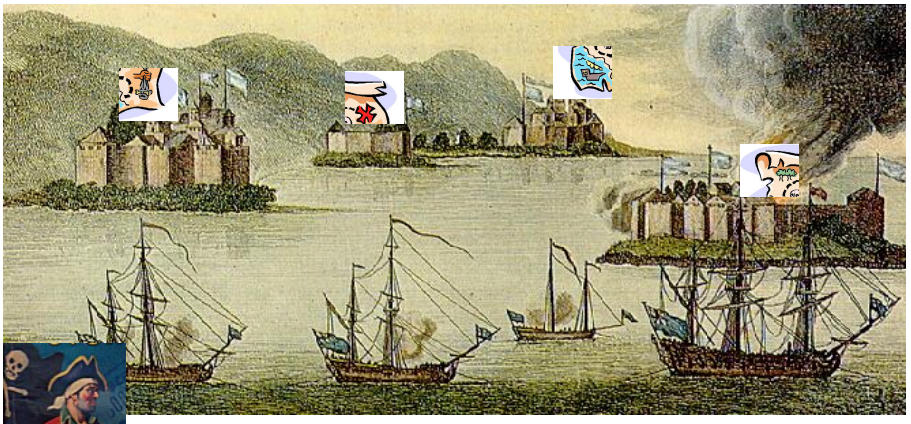
Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

Replicação



Fragmentação



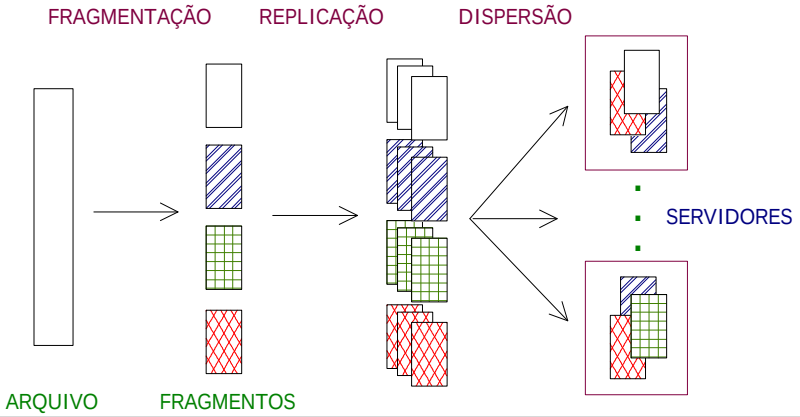
disponibilidade, integridade, confidencialidade
armazenamento eficiente

Fragmentação – FRS

- Trabalho seminal: Fraga e Powell 85 – FRS
  - ☞ *fragmentation-redundancy-scattering*
  - ☞ não fornece confidencialidade...
    "reduz o significado da informação disponível a um intruso"

FRS

- armazenamento de um arquivo F em N servidores



FRS (cont)

- Integridade – 2 soluções para detectar fragmentos corrompidos:
  - junta-se um MAC a cada fragmento
  - lê-se diversas cópias de cada fragmento; votação (reduz resistência)
- Servidores de segurança (TI)
  - localização dos arquivos
  - autorização de acesso aos arquivos

- Este trabalho trata de muitos dos problemas que foram depois estudados em TI

Otimização do espaço e confidencialidade

- Como melhorar o espaço ocupado por um arquivo nos servidores?
  - **códigos de apagamento** (erasure codes)
    - usam informação redundante para permitir recuperar os dados se uma parte for *apagada*
  - código de apagamento-(k,N)
    - o arquivo é dividido em **N** fragmentos
    - são necessários **k** fragmentos para o reconstruir
  - Usados para obter confidencialidade num artigo de Cachin e Tessaro (SRDS'05)

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

Fragmentação

se tiver muito tempo o pirata pode conseguir vários fragmentos...

Recuperação proativa

periodicamente faz-se uma *renovação*

Processamento de erros e TI

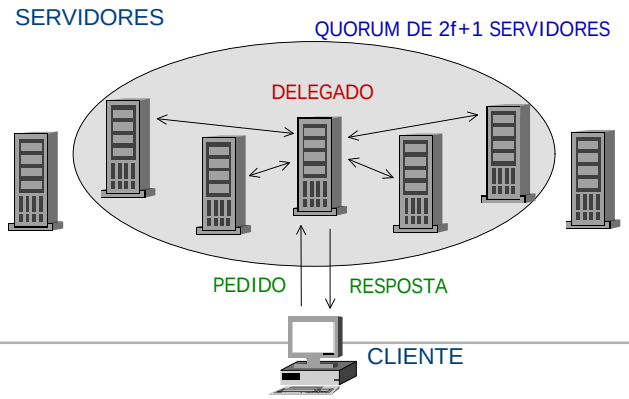
- processamento de erros faz parte da *tolerância a faltas*
- técnica mais intuitiva em TI:
  - detectar intrusões + renovar servidor
  - mas IDSs geram muitos falsos positivos e falsos negativos...
- **recuperação proativa**:
  - periodicamente renova-se cada servidor
  - a premissa habitual passa a ser *não podem falhar mais do que **f** servidores em cada período* – **janela de vulnerabilidade**

COCA

- *Cornell On-line Certification Authority* (OASIS)
- autoridade de certificação *on-line*
  - ☞ fornece certificados com associações {nome, chave pública}
  - ☞ update: criar, actualizar ou invalidar certificados
  - ☞ query: obter o certificado correspondente a um nome
- baseado em **quorums de disseminação**
  - ☞ ***N = 3f+1*** ***|Q| = 2f+1***
- usa esquema de *assinatura de limiar-(k,N)*
  - ☞ todos os clientes e servidores têm a chave pública
  - ☞ a chave privada do serviço está dividida pelos servidores e são necessários ***k=f+1*** para assinarem um certificado

COCA - funcionamento

- pedido enviado ao delegado que reenvia para um quorum (bastam f+1 para assinar certificado)
- delegado malicioso pode exigir retransmissão (p/ f+1)



COCA – recuperação proativa

- três operações:
  - renovar partes da chave privada de cada servidor
  - repor código do servidor se corrompido
  - repor estado do servidor se corrompido
- renovar partes da chave privada
  - se o atacante apanhasse ***f+1*** poderia personificar o serviço...
  - *protocolo proativo de partilha de segredos APSS*
    - periodicamente gera novas partes
    - a chave privada mantém-se a mesma
    - a chave privada nunca é materializada num servidor
- sistema CODEX: versão do COCA para armazenamento de dados

Recuperação proativa em sistemas assíncronos?

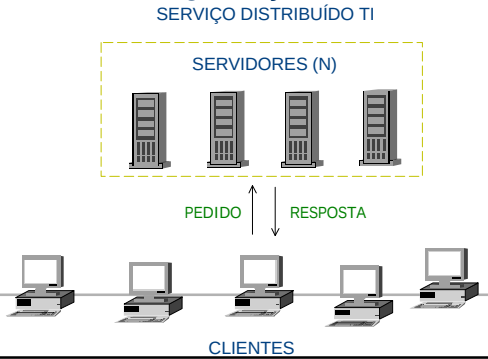
- modelo assíncrono
  - não estabelece hipóteses temporais
  - logo não cria vulnerabilidades desse tipo
- *não se pode fazer recuperação proativa de forma segura (safe) em sistemas assíncronos*
  - resultado recente
  - Sousa, Neves, Veríssimo 2005
- solução: sistema assíncrono + componente distribuída segura que encapsula a sincronia necessária (wormhole)

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

Arquiteturas

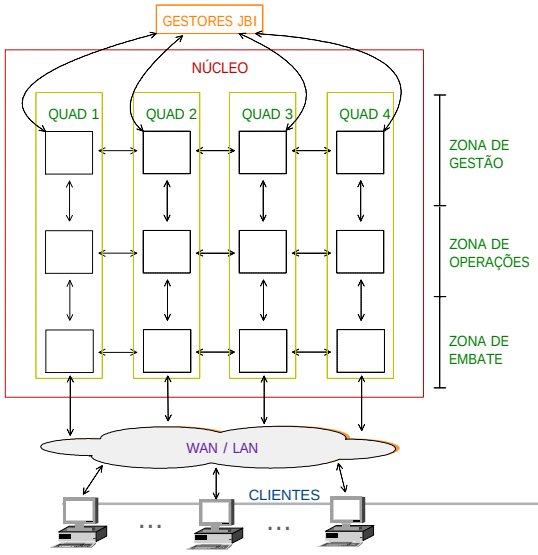
- vimos quatro tipos se soluções:
  - ☞ RME, quoruns, fragmentação, recuperação proativa
- todas baseadas numa arquitetura simples
- arquiteturas mais complicadas juntam essas soluções com outras de TI e segurança



DPASA

- *Designing Protection and Adaptation into a Survivability Architecture* (OASIS Dem/Val)
- usada em aplicação da força aérea americana: *Joint Battlespace Infosphere (JBI)*
  - ☞fornece serviços de publicação-subscrição-interrogação (PSQ) de forma a que os clientes possam trocar informação sob a forma de objetos de informação (OIs).
- o IT-JBI consiste num núcleo central que fornece serviços de comunicação a um conjunto de clientes

Arquitetura DPASA (simplificada)



- SOs diferentes nos *quads:*
  - ☞SELinux, Solaris, Windows XP, Windows 2000
- adaptadores de rede 3COM
  - ☞firewall embutida e VPN
- zona embate
  - ☞proxy de acesso c/interface protocolos e middleware dos clientes (em VPN)
- zona operações
  - ☞processamento, detecção OIs corrompidos
- zona de gestão
  - ☞correlaciona e filtra alarmes
- resistiu a ataques de 2 red teams (conhecimento total) excepto alguns ataques negação serviço

Organização da palestra

- Conceitos básicos de TI
- Replicação
- Fragmentação
- Recuperação proativa
- Arquiteturas e sistemas
- Conclusão

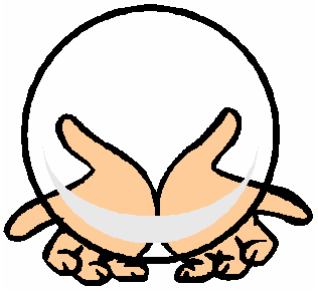
A TI permite melhorar a segurança?

- sim...
- uma arquitetura TI completa inclui numerosos componentes e mecanismos que vêm da *segurança* clássica, não são nenhuma novidade da TI (p.ex., IT-JBI)
- a possibilidade de multiplicar por N (ou ***f+1*** ou ...) a dificuldade de atacar e controlar um serviço traz claramente mais segurança

A TI está pronta para ser usada?

- sim
- existem lacunas que podem ser pesquisadas
- a TI pode ser tornada muito mais prática de usar
- mas as soluções que vimos são reais e utilizáveis já hoje

Em que tipo de aplicações será realmente usada?

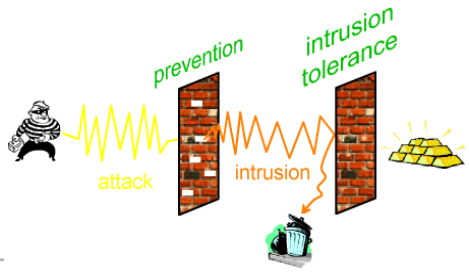


Aplicações de TI?

- soluções complexas e caras para aplicações críticas
  - protecção de infraestruturas críticas (projeto CRUTIAL)
  - aplicações financeiras (rede SWIFT)
  - aplicações militares (IT-JBI)
  - há alguns anos a chave de raiz do sistema SET da MasterCard/VISA se encontrava distribuída por diversas empresas diferentes usando criptografia de limiar
- soluções simples poderão ser usadas para tornar mais seguros componentes/sistemas de menor custo
  - ex: soluções de armazenamento de dados em rede (NAS, SAN) TI comerciais

Perguntas?

- Página pessoal
  http://www.di.fc.ul.pt/~mpc
- Grupo Navigators:
  http://www.navigators.di.fc.ul.pt/
- Email:
  mpc@di.fc.ul.pt



Para saber mais...

- Sobre tolerância a intrusões no geral
  - M. P. Correia. **Serviços Distribuídos Tolerantes a Intrusões: resultados recentes e problemas abertos**. V SBSeg - Livro Texto dos Minicursos, 2005
  - P. Veríssimo and N. F. Neves and M. Correia. **Intrusion-Tolerant Architectures: Concepts and Design.** In *Architecting Dependable Systems*, LNCS 2677, Springer, 2003
- Artigos em revistas
  - M. Correia, N. F. Neves, L. C. Lung, P. Veríssimo. **Worm-IT - A Wormhole-based Intrusion-Tolerant Group Communication System**. Journal of Systems & Software, Elsevier, 2006. to appear
  - M. Correia, N. F. Neves, P. Veríssimo. **From Consensus to Atomic Broadcast: Time-Free Byzantine-Resistant Protocols without Signatures**. Computer Journal. vol. 41, n. 1, pp 82-96, January 2006
  - N. F. Neves, M. Correia, P. Veríssimo. **Solving Vector Consensus with a Wormhole**. IEEE Transactions on Parallel and Distributed Systems, Volume 16, Issue 12, Dec. 2005
  - M. Correia, N. F. Neves, L. C. Lung, P. Veríssimo. **Low Complexity Byzantine-Resilient Consensus**. Distributed Computing, vol. 17, n. 3, pp. 237--249, March 2005.
- Artigos recentes em conferências
  - P. Veríssimo, N. F. Neves and M. Correia. **CRUTIAL: The Blueprint of a Reference Critical Information Infrastructure Architecture**. *In CRITIS'06 1st International Workshop on Critical Information Infrastructures Security.* August 30 - September 2, 2006.
  - H. Moniz and N. F. Neves and M. Correia and P. Veríssimo. **Experimental Comparison of Local and Shared Coin Randomized Consensus Protocols**. *27th IEEE Symposium on Reliable Distributed Systems*. October 2006
  - H. Moniz and N. F. Neves and M. Correia and P. Veríssimo. **Randomized Intrusion-Tolerant Asynchronous Services**. *International Conference on Dependable Systems and Networks (DSN),* June 2006.
  - N. F. Neves and J. Antunes and M. Correia and P. Veríssimo and R. Neves. **Using Attack Injection to Discover New Vulnerabilities**. *International Conference on Dependable Systems and Networks (DSN),* June 2006.
  - A.N.Bessani and M.Correia and J.S.Fraga and L.C.Lung. **Sharing Memory between Byzantine Processes using Policy-Enforced Tuple Spaces**. *26th International Conference on Distributed Computing Systems,* July 2006.