Sistemas Críticos

UNIP - Araraquara

Curso: Ciências da Computação

Disciplina: Qualidade de Software

Profº: João Paulo Moreira dos Santos

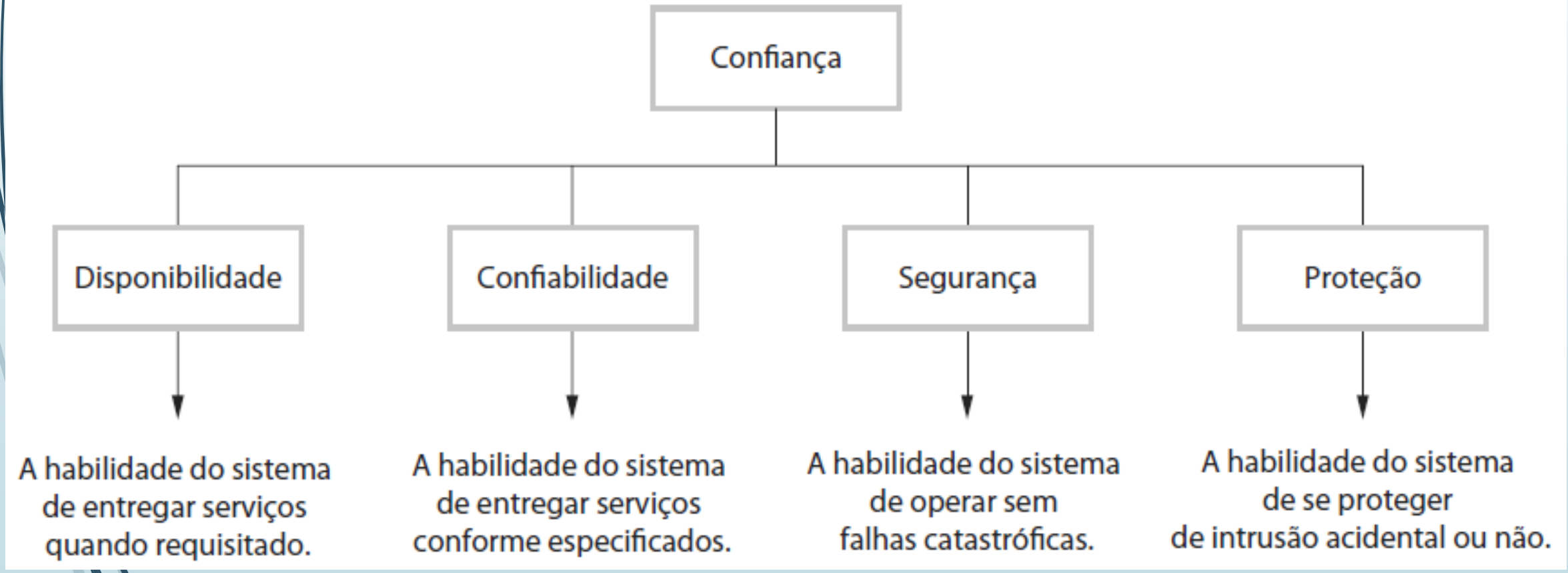
Definição

- São sistemas técnicos ou sociotécnicos dos quais as pessoas ou os negócios dependem. Caso esses sistemas falhem, os problemas serão gravíssimos.

- Existem 3 tipos de sistemas críticos:
  - **Sistemas críticos de segurança:** Sistemas cuja a falha pode resultar em prejuízo, danos sérios ou ao ambiente.
  - **Sistemas críticos de missão:** Sistemas cuja falha pode causar problemas em objetivos, missões etc.
  - **Sistemas críticos de negócios:** Sistemas em cuja a falha pode acarretar perdas financeiras em algum negócio.

Sistemas Críticos

- A propriedade mais importante de um sistema crítico é a **confiança**. O termo é abrangido para relacionar a atributos como:



Confiança no Sistema

- **Disponibilidade:**
  - Probabilidade de que o sistema esteja pronto e em execução, capaz de fornecer serviços úteis a qualquer instante
- **Confiabilidade:**
  - Probabilidade de que o sistema, forneça corretamente os serviços, conforme esperado pelo usuário.
- **Segurança:**
  - Análise da probabilidade de um sistema causar danos para pessoas ou para o ambiente.
- **Proteção:**
  - Análise da probabilidade de que um sistema possa resistir a intrusões acidentais ou deliberadas.

Sistemas Críticos

- Razões para que a confiança seja prioridade mais importante:
  - Sistemas não confiáveis, inseguros ou desprotegidos são frequentemente rejeitados por seus usuários.
  - Os custos de falha de sistema podem ser muito altos.
  - Sistemas não confiáveis podem causar perda de informações.

Sistemas Críticos

- Por serem tão importantes, sistemas críticos em geral são desenvolvidos usando técnicas já consagradas em vez de técnicas mais recentes.
- Um motivo pelo qual os métodos convencionais são mais usados, é que eles ajudam a reduzir a quantidade dos testes necessários.
- Em sistemas críticos, custos de validação são geralmente muito altos, representando mais de 50% dos custos totais de desenvolvimento do sistema.

Falhas do Sistema

- Quando ocorrem falhas nesse tipo de sistemas, geralmente é necessário que pessoas assumam o controle, para poder contornar as dificuldades.

- Existem três tipos de falhas que podem ocorrer em sistemas críticos:

  - O hardware pode falhar.
  - O software pode falhar.
  - Os operadores podem falhar.

Falhas do Sistema

- Distinção entre Erro, Defeito e Falha:

Uma pessoa comete um **erro**...

... que cria um **defeito** no software...

... que pode causar uma **falha** na operação

Falhas do Sistema

- A distinção entre erros, defeitos e falhas, ajuda a identificar três abordagens complementares usadas para melhorar a confiabilidade de um sistema:

1. **Prevenção de defeitos:** Técnicas de desenvolvimento são usadas para minimizar a possibilidade de erros humanos antes que eles resultem na introdução de defeitos de sistema.
    - Exemplo dessas técnicas incluem evitar construções de linguagem de programação propensas a erro, como ponteiros e uso da análise estática para detectar anomalias de programa.

Falhas do Sistema

- A distinção entre erros, defeitos e falhas, ajuda a identificar três abordagens complementares usadas para melhorar a confiabilidade de um sistema:

2. **Detecção e remoção de defeitos:** O uso de técnicas de verificação e validação aumenta as chances de detecção e remoção de defeitos antes de o sistema ser usado.

  - Testes e depuração sistemáticos são exemplos de técnicas de detecção de defeitos.

Falhas do Sistema

- A distinção entre erros, defeitos e falhas, ajuda a identificar três abordagens complementares usadas para melhorar a confiabilidade de um sistema:

3. **Tolerância a defeitos:** São as técnicas que asseguram que os defeitos em um sistema não resultam em falhas de sistemas.

  - A incorporação de recursos de autoverificação em um sistema e o uso de módulos redundantes de sistemas são exemplos de técnicas de tolerância a defeitos.

Segurança

- Os sistemas críticos de segurança são os sistemas no quais é essencial que a operação seja sempre segura.
  - Isso significa que o sistema não pode causar danos as pessoas ou ao meio ambiente mesmo que o sistema venha falhar.

- Exemplos:

Segurança

- Atualmente os sistemas são tão complexos que não podem ser controlados só por hardware e o controle de software passa a ser essencial.

- Um controle de software é essencial pela necessidade de gerenciar um grande número de sensores e atuadores com leis de controle complexos.

Segurança

- O software crítico de segurança divide-se em duas classes:

- **Software crítico de segurança primária:** Esse é um software embutido como um controlador em um sistema. O mau funcionamento do software pode causar mau funcionamento do hardware, o que resulta em danos às pessoas ou ao ambiente.

- **Software crítico de segurança secundária:** Esse é um software cuja falha resulta em defeitos em outros sistemas que podem ameaçar as pessoas.

Segurança

- A confiabilidade e a segurança de um sistema estão relacionadas, mas um sistema confiável pode ser inseguro e vice-versa.

- O software pode comportar-se de tal forma que o resultado do comportamento do sistema cause um acidente.

Segurança

- Existem quatro razões pelas quais os sistemas de software que são confiáveis não são necessariamente seguros:

1. Nós nunca podemos estar 100% certos de que um sistema de software seja livre de defeitos ou tolerante a defeitos.
   - Defeitos não detectados podem ficar adormecidos por um longo tempo e falhas de software podem ocorrer após vários anos de funcionamento confiável.
2. A especificação pode ser incompleta, sem a descrição do comportamento requerido do sistema em algumas situações críticas.
   - Uma elevada porcentagem de mau funcionamentos de sistema resulta da especificação, e não de erros de projeto.

Segurança

3. Maus funcionamentos de hardware podem levar o sistema a se comportar de forma imprevisível, bem como apresentar o software com um ambiente imprevisto.
   - Quando componentes estão perto de falha física, eles podem se comportar de forma errática e gerar sinais que estão fora dos intervalos que podem ser manipulados pelo software.

4. Os operadores de sistema podem gerar entradas que por si só não são erradas, mas em algumas situações podem causar um mau funcionamento de sistema.

Segurança

- A chave para garantir a segurança é assegurar que os acidentes não ocorram e/ou que as consequências de um acidente sejam mínimas. Isso pode ser alcançado de três maneiras complementares:

1. **Prevenção de perigos:** O sistema é projetado de modo que os riscos sejam evitados.
2. **Detecção e remoção de perigos:** O sistema é projetado de modo que os perigos sejam detectados e removidos antes que resultem em um acidente.
3. **Limitação de danos:** O sistema pode incluir recursos de proteção que minimizem os danos que possam resultar em acidente.

Segurança

- Quase todos os acidentes são resultado de combinações de mau funcionamentos.
- Em todos os casos, é importante manter um senso de proporção sobre a segurança do sistema.
- É impossível fazer um sistema 100% seguro, e a sociedade precisa decidir se consequências de um acidente ocasional valem os benefícios do uso de tecnologias avançadas ou não.

Proteção

- A segurança é um atributo do sistema que reflete sua capacidade de se proteger de ataques externos, sejam acidentais ou deliberados.
- Esses ataques são possíveis porque a maioria dos computadores de uso geral está em rede e é, portanto, acessível a estranhos.
- Exemplos de ataques podem ser a instalação de vírus e cavalos de Troia, o uso não autorizado de serviços de sistema ou a modificação não autorizada de um sistema ou seus dados.

Proteção

- Se você quer um sistema realmente seguro, é melhor não o conectar à Internet. Assim, seus problemas de proteção serão limitados a garantir que usuários autorizados não abusem do sistema.

- Na prática, porém, existem enormes benefícios no acesso à rede, não sendo rentável à maioria dos grandes sistemas desconectar-se da Internet.

Proteção

- Para alguns sistemas, a proteção é a dimensão mais importante da confiança de sistema.
  - Sistemas militares, sistemas de comércio eletrônico e sistemas que envolvem processamento e intercâmbio de informações confidenciais, por exemplo, devem ser projetados de modo a alcançar um elevado nível de proteção.
  - Se um sistema de reserva de passagens aéreas não estiver disponível, por exemplo, esse inconveniente pode causar alguns atrasos na emissão de bilhetes; ou, ainda, se o sistema não tiver proteção, o invasor pode, em seguida, apagar todas as reservas, tornando praticamente impossível às operações normais continuarem.

Proteção

- Existe alguns termos importantes referentes a proteção:

Termo	Definição
Ativo	Algo de valor que deve ser protegido. O ativo pode ser o próprio sistema de software ou dados usados por esse sistema.
Exposição	Possíveis perdas ou danos a um sistema de computação. Pode ser perda ou dano aos dados, ou uma perda de tempo e esforço, caso seja necessária a recuperação após uma brecha de proteção.
Vulnerabilidade	A fraqueza em um sistema computacional, que pode ser explorada para causar perdas ou danos.
Ataque	Uma exploração da vulnerabilidade de um sistema. Geralmente, vem de fora do sistema e é uma tentativa deliberada de causar algum dano.
Ameaças	Circunstâncias que têm potencial para causar perdas ou danos. Você pode pensar nisso como uma vulnerabilidade de um sistema submetido a um ataque.
Controle	Uma medida de proteção que reduz a vulnerabilidade do sistema. A criptografia é um exemplo de controle que reduz a vulnerabilidade de um sistema de controle de acesso fraco.

Proteção

- Em qualquer sistema de rede, existem três principais tipos de ameaças à proteção:

1. **Ameaças à confidencialidade do sistema e seus dados:** Essas ameaças podem divulgar informações para pessoas ou programas não autorizados a acessarem-nas.

2. **Ameaças à integridade do sistema e seus dados:** Essas ameaças podem danificar o software ou corromper seus dados.

3. **Ameaças à disponibilidade do sistema e seus dados:** Essas ameaças podem restringir, para usuários autorizados, acesso ao software ou a seus dados.

Proteção

- Essas ameaças são, naturalmente, interdependentes.
  - Se um ataque tornar o sistema indisponível, você não será capaz de atualizar as informações que mudam com tempo. Isso significa que a integridade do sistema pode estar comprometida.
  - Se um ataque for bem-sucedido e a integridade do sistema for comprometida, então pode ser necessário parar para reparar o problema. Portanto, a disponibilidade do sistema ficará reduzida.

Proteção

- Na prática, a maioria das vulnerabilidades em sistemas sociotécnicos resulta de falhas humanas e não de problemas técnicos.
  - As pessoas escolhem senhas fáceis de adivinhar ou anotam suas senhas em lugares onde podem ser encontradas.
  - Os administradores de sistema cometem erros na configuração de controle de acesso ou arquivos de configuração, e os usuários não instalam ou não usam softwares de proteção.

Proteção

- Na prática, a maioria das vulnerabilidades em sistemas sociotécnicos resulta de falhas humanas e não de problemas técnicos.
  - As pessoas escolhem senhas fáceis de adivinhar ou anotam suas senhas em lugares onde podem ser encontradas.
  - Os administradores de sistema cometem erros na configuração de controle de acesso ou arquivos de configuração, e os usuários não instalam ou não usam softwares de proteção.

Proteção

- No entanto, precisamos ter muito cuidado ao classificar o problema como um erro de usuário.

- Muitas vezes, os problemas humanos refletem decisões pobres, tomadas durante o projeto de sistema, por exemplo, a alteração frequente de senhas (que exige que os usuários anotem suas senhas) ou mecanismos de configurações complexos.

Proteção

- No entanto, precisamos ter muito cuidado ao classificar o problema como um erro de usuário.

- Muitas vezes, os problemas humanos refletem decisões pobres, tomadas durante o projeto de sistema, por exemplo, a alteração frequente de senhas (que exige que os usuários anotem suas senhas) ou mecanismos de configurações complexos.

Proteção

- Os controles que você pode colocar em prática para melhorar a proteção de sistema são comparáveis àqueles de confiabilidade e segurança:

1. **Prevenção de vulnerabilidade**;
2. **Detecção e neutralização de ataques**; e
3. **Limitação de exposição e recuperação**.

Proteção

1. **Prevenção de vulnerabilidade**: controles que se destinam a assegurar que os ataques não sejam bem-sucedidos.
   - Por exemplo, sistemas militares sensíveis não estão ligados às redes públicas, de modo a tornar impossível o acesso externo.
2. **Detecção e neutralização de ataques**: controles que visam detectar e repelir os ataques.
   - Esses controles incluem, em um sistema, funcionalidade que monitora sua operação e verifica padrões incomuns de atividade. Se tais padrões forem detectados, uma ação poderá ser tomada, como desligar partes do sistema, restringir o acesso a determinados usuários etc.
3. **Limitação de exposição e recuperação**: controles que apoiam a recuperação de problemas.
   - Podem variar desde estratégias de backup automatizadas e 'espelhamento' de informações para políticas de seguro que cubram os custos associados a um ataque bem-sucedido ao sistema.

Proteção

- Sem um nível razoável de proteção, não podemos estar confiantes quanto à disponibilidade, à confiabilidade e à segurança de um sistema.
- Métodos para a certificação de disponibilidade, confiabilidade e proteção assumem que um software operacional seja o mesmo software originalmente instalado.
- Se o sistema tiver sido atacado e o software tiver sido comprometido de alguma maneira (por exemplo, se o software foi modificado para incluir um worm), os argumentos de confiabilidade e de proteção não são mais válidos.

Proteção

- Erros no desenvolvimento de um sistema podem causar brechas de proteção.

- Se um sistema não responde às entradas inesperadas ou se limites de vetor não são verificados, os invasores podem, em seguida, explorar essas fraquezas para obter acesso ao sistema.