

O que é Configuration Management?

DigitalHouse >
Coding School



**Certified Tech
Developer**
The Ultimate Degree

Índice

1. [Introdução](#)
2. [Traditional Configuration Management](#)
3. [Problemáticos](#)
4. [cattle-vs-pet](#)

1 | Introdução



Configuration Management e **Change Management** são dois dos processos fundamentais do conjunto de metodologias conhecido como ITIL.



Gestão de Mudanças

A ITIL descreve o gerenciamento de mudanças como o processo de controlar e gerenciar uma mudança ao longo de seu ciclo de vida com o objetivo de minimizar o risco.

O que é uma mudança?

De acordo com a ITIL, uma mudança é a modificação ou remoção de qualquer coisa que possa afetar direta ou indiretamente os serviços. Basicamente, qualquer mudança na infraestrutura de TI de uma organização pode afetar as operações da organização.

Exemplos de uma mudança:

Substituição de hardware, instalação de software em um servidor ou modificação de uma configuração de sistema que altere seu comportamento.

Configuration Management

É o processo que nos permite gerenciar as alterações de configuração de nossos ativos de TI, sejam eles software ou hardware, permitindo que a organização mantenha um registro histórico e, por sua vez, aplique controles - como um processo de aprovação de alterações que atendam a determinadas características.

Cada um dos ativos de TI no contexto desse processo é conhecido como **itens de configuração (IC)** e são armazenados no que é chamado de **SGBD** (banco de dados de gerenciamento de configuração).

Configuration Management + Change Management

Ambos os processos se complementam, pois a **Change Management** contribui para a governança das mudanças que são realizadas no parque tecnológico de uma organização, e a **Configuration Management** nos permite gerenciar um banco de dados (SGBD) com as informações de nossos ativos (ICs) e um histórico das mudanças feitas para cada um deles.

Dessa forma, se ocorrer uma falha em um de nossos ativos que nos obrigue a restaurá-lo, por exemplo, uma falha crítica em um servidor, o processo de gerenciamento de configuração nos fornece uma visão de todas as alterações ocorridas nesse ativo desde sua criação e, assim, nos permitem reproduzir cada uma das alterações e devolvê-la ao estado anterior à falha.

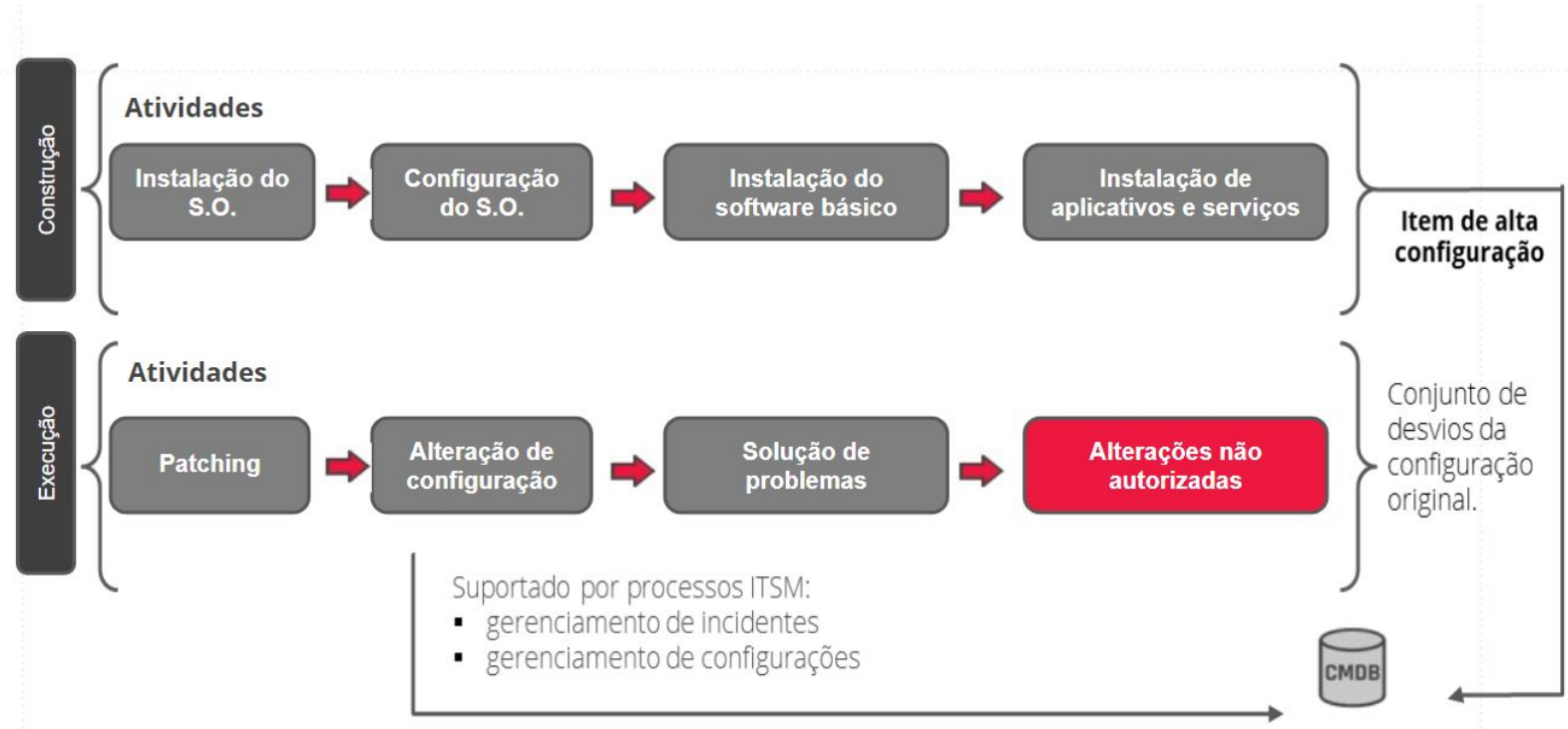
2 | Traditional Configuration Management

Como isto funciona na prática?

Vamos colar o ciclo de vida de um servidor como exemplo.
Pode ser dividido em 2 etapas:

- A etapa de **construção** do servidor: um conjunto de etapas que podem ser automatizadas, documentadas ou simplesmente conhecimento comum de TI, que permitem que um servidor seja lançado e registrado no **SGBD** como um novo **IC**.
- Seu ciclo de vida, que podemos chamar de fase de **execução**: tudo o que acontece em relação ao ativo e as mudanças que podem ocorrer nele.

Mapa de processo tradicional



Etapa de construção de um servidor

Instalação do SO

O processo de instalação do sistema operacional no hardware ou na máquina virtual.

Configuração do SO

O conjunto de configurações definidas pela organização para que o servidor possa fazer parte do ecossistema.

Instalando o software básico

É provável que a organização de TI tenha determinado que software adicional deve ser instalado para que o servidor funcione corretamente no ecossistema. Estes podem incluir: antivírus, software IDS/IPS, agentes de backup, etc.

Instalação de aplicativos e serviços

Por fim, durante o processo de construção do servidor, deve ser instalado e configurado o software ou serviço que dará sentido à sua existência. Por exemplo, se o servidor pretende ser um servidor web, não se trata apenas de instalar o servidor web (IIS, NGINX, Apache, etc), mas também de implantar o site que o hospedará.

Fase de execução de um servidor

Patching

O processo de instalação da atualização do sistema operacional, dependendo do sistema operacional é a cadência com que esse processo será executado. De um modo geral, as organizações optam por executar instalações de atualização mensais. Este processo deve ser refletido através do processo de gerenciamento de mudanças.

Alterações de configuração

Qualquer alteração de configuração de hardware ou sistema operacional que responda a uma necessidade. Estes devem ser refletidos como parte do processo de gerenciamento de mudanças.

Solução de problemas

Qualquer incidente que tenha produzido uma falha no sistema reduzindo ou cancelando sua operação. Requer intervenção do administrador para restaurar o nível de serviço. Estes devem ser refletidos como parte do processo de gerenciamento de incidentes.

Alterações não autorizadas

Alerta! Qualquer alteração ou atividade feita no servidor que não esteja documentada em um ticket, seja como alteração ou incidente.

3 | Problemáticos

Configuration Management + Change Management, é viável?

É possível reconstruir um ativo para o mesmo estado em que estava antes da falha que nos obrigou a restaurá-lo? Ser capaz de fazer isso com sucesso depende de muitas variáveis:

- A disciplina que tivemos deve realmente refletir cada uma das mudanças feitas no ativo durante seu ciclo de vida.
- Ficará a critério do administrador responsável pela restauração quais alterações devem ser reproduzidas e quais podem ser ignoradas.
- Erro humano, podemos cometer um erro ao tentar reconstruir o ativo. Toda tarefa manual leva a erros. E quanto mais antigo for um ativo, maior o número de alterações que temos que aplicar, conseqüentemente, maior a possibilidade de errar.
- A existência de alterações não autorizadas.

“

Uma alteração não autorizada implementada com sucesso é mais **perigosa** do que uma que não é, uma vez que produz uma configuração eficaz sem deixar evidências disso.

”



4 | cattle-vs-pet

cattle-vs-pet

Quando se trata de gerenciamento de configuração, há uma tendência de fazer a diferença entre 'Animais de estimação' e 'Gado'. Por quê?

Em termos gerais, diz-se que nos processos tradicionais, aqueles em que encontramos os problemas mencionados anteriormente, cada servidor gerido individualmente torna-se um animal de estimação, com tratamento personalizado.

Já nos processos modernos de gerenciamento de servidores, eles são gerenciados como um todo, com uma visão industrial e abrangente do parque tecnológico. É adotada uma abordagem que nos permite implementar correções rápidas e substituir componentes com falha, em vez de gastar tempo tentando resolver problemas complexos. Semelhante ao que acontece na pecuária onde o gado é manejado como um todo.

Configuração como código para o resgate!

Nos processos modernos de gerenciamento de configuração, a capacidade de definir a configuração de um servidor como código (Configuration as Code ou CaC) é essencial.

- Gerenciar o parque tecnológico como gado não seria possível sem o uso de um sistema de gerenciamento de configuração.
- Qualquer coisa definida como código pode ser automatizada.
- O código pode ser testado.
- O CaC é um passo antes de habilitar Self-Healing e Self-Remediation (duas práticas do mundo da infraestrutura moderna que permitem a implementação de processos de autoreparação).

Configuração como código para o resgate!

- É compatível com o processo de gerenciamento de mudanças ITIL, pois as modificações não acontecem nos ativos, mas em um repositório que suporta versionamento. Assim, as alterações podem ser testadas e implantadas em ambientes de baixo custo e, em seguida, aplicadas na produção.
- Ele não substitui o processo de gerenciamento de configuração ITIL, eles até compartilham o mesmo nome.

Os desafios de adotar a Configuration as Code

A transição de processos tradicionais para processos modernos pode nos levar a enfrentar vários desafios:

- Temos que entender se realmente faz sentido fazer a transição para esses processos.
- Entenda até onde queremos ir com a implementação do CaC, vamos modernizar os sistemas existentes? Ou adotaremos a prática apenas para novos sistemas?
- Abandonar velhas formas de trabalho pode apresentar resistência.
- Talvez precisemos repensar a forma como os servidores são construídos na organização.

Ferramentas de Gerenciamento de Configuração







PowerShell DSC



Detalhamento de cada tecnologia:

https://docs.google.com/document/d/14x876UstcKmkaQAxOY2ef3BSkm_7rEDjx1ahqIJ_t4w/edit

Ferramentas de Gerenciamento de Configuração

FERRAMENTAS	PRÓS	CONTRAS
 CHEF	<ul style="list-style-type: none">• Muito estável e confiável.• Comunidade ativa.• Uma das soluções mais flexíveis.	<ul style="list-style-type: none">• Curva de aprendizado lenta.• Falta push, então não tem nenhuma ação imediata sobre as mudanças. O processo de pull segue um cronograma especificado.
 puppet	<ul style="list-style-type: none">• Forte automação de compilação e ferramentas de relatórios.• Suporte ativo da comunidade.• Interface web intuitiva para cuidar de muitas tarefas.	<ul style="list-style-type: none">• Pode ser difícil para novos usuários que precisam aprender Puppet DSL ou Ruby.• Não é a melhor solução disponível em relação à escalabilidade.
 ANSIBLE	<ul style="list-style-type: none">• Sistema mais simples em relação a instalação e manutenção.• Suporta modelos push e pull• <u>Sem</u> agentes.	<ul style="list-style-type: none">• A comunicação SSH fica mais lenta em ambientes com muitos servidores.• A sintaxe entre os componentes de script pode variar.
 SALTSTACK	<ul style="list-style-type: none">• Uso fácil e direto após a instalação e configuração inicial.• Eficaz para ambientes com alta escalabilidade e resilientes.	<ul style="list-style-type: none">• Instalação inicial complexa.• A documentação não é bem gerenciada.• Não é a melhor opção para sistemas operacionais diferentes do Linux.

Detalhamento de cada tecnologia:

https://docs.google.com/document/d/14x876UstcKmkaQAxOY2ef3BSkm_7rEDjx1ahqIJ_t4w/edit

DigitalHouse>
Coding School