



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2017년04월19일

(11) 등록번호 10-1728579

(24) 등록일자 2017년04월13일

- (51) 국제특허분류(Int. Cl.)
G06F 17/00 (2006.01) *G06F 15/16* (2006.01)
G06F 9/44 (2006.01)
- (21) 출원번호 10-2012-7023636
- (22) 출원일자(국제) 2011년03월04일
심사청구일자 2016년02월04일
- (85) 번역문제출일자 2012년09월10일
- (65) 공개번호 10-2013-0029047
- (43) 공개일자 2013년03월21일
- (86) 국제출원번호 PCT/US2011/027307
- (87) 국제공개번호 WO 2011/112473
국제공개일자 2011년09월15일
- (30) 우선권주장
12/721,579 2010년03월11일 미국(US)
- (56) 선행기술조사문헌
US20080028048 A1
WO2008013897 A1

- (73) 특허권자
마이크로소프트 테크놀로지 라이선싱, 엘엘씨
 미국 워싱턴주 (우편번호 : 98052) 레드몬드 원
 마이크로소프트 웨이
- (72) 발명자
샤 손 케이
 미국 워싱턴주 98052-6399 레드몬드 원 마이크로
 소프트 웨이 엘엘씨에이 - 인터내셔널 패이턴츠 마
 이크로소프트 코포레이션
- 스리다르 프라산나 에이치**
 미국 워싱턴주 98052-6399 레드몬드 원 마이크로
 소프트 웨이 엘엘씨에이 - 인터내셔널 패이턴츠 마
 이크로소프트 코포레이션
- (뒷면에 계속)
- (74) 대리인
제일특허법인

전체 청구항 수 : 총 18 항

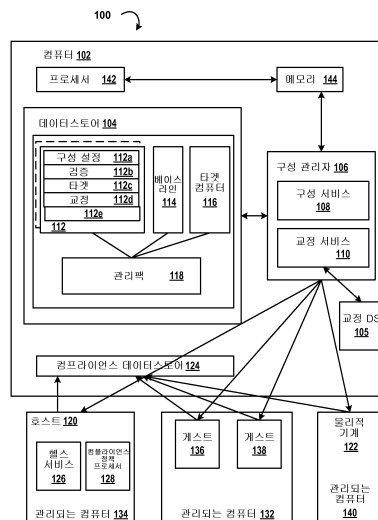
심사관 : 홍경아

(54) 발명의 명칭 **효과적인 구성 드리프트 관리 기법**

(57) 요약

구성 드리프트는 컴퓨터 또는 서비스가 희망 구성으로부터 이탈하도록 하는 시간에 따라 이루어지는 변화를 지칭한다. 기계 그룹의 구성 드리프트는 구성 의도를 정의함으로써 관리될 수 있다. 관련된 구성 규칙의 집합으로 이루어진 구성 베이스라인을 정의함으로써 의도가 정의된다. 구성 규칙은 설정, 그리고 더 자세한 수준에서 비합치의 보고를 가능하게 하는 여하한 관리된 엔티티일 수 있는 타겟을 포함한다. 구성 베이스라인은 하나 이상의 잘 구성된 컴퓨터로부터 구성 규칙을 판독함으로써 완성될 수 있다. 구성 드리프트는 실제 값을 구성 베이스라인 값과 비교함으로써 평가되고, 기계 수준이 아니라 관리된 엔티티 수준에서 보고된다. 구성 베이스라인과 합치하는 상태로 컴퓨터를 돌려놓는 교정은 요청에 따라 수행될 수 있다.

대표도



(72) 발명자

피니간 제임스 피

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 패이턴츠 마이크로소프트 코포레이션

파사사라시 스리바트산

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 패이턴츠 마이크로소프트 코포레이션

굿맨 알란 에이치

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 패이턴츠 마이크로소프트 코포레이션

명세서

청구범위

청구항 1

시스템으로서,

메모리 및 프로세서를 포함하고,

상기 메모리는 상기 프로세서로 하여금

구성 규칙의 집합(a collection of configuration rules)을 정의함으로써 복수의 컴퓨터에 대한 구성 베이스라인(configuration baseline)을 정의하고 -상기 구성 규칙의 집합의 각각은, 프로그램, 서비스, 애플리케이션, 데이터베이스, 운영체제 또는 웹사이트를 포함하는 관리된 엔티티(managed entity)를 포함하는 타겟과 연관되고, 상기 구성 규칙의 집합 중 하나의 구성 규칙은 타겟을 구성 베이스라인에 합치하는 상태로 되돌리기 위해 상기 구성 규칙에 대해 구성 드리프트(configuration drift)를 교정하도록 자동으로 실행하는 명령어 세트를 포함하며, 상기 구성 베이스라인은 다수의 기존 컴퓨터 각각으로부터 판독된 적어도 하나의 구성 설정을 결합함으로써 생성됨- ,

상기 구성 규칙의 집합을 포함하는 상기 정의된 구성 베이스라인에 기초하여 관리 팩(management pack)을 생성하며,

상기 관리 팩을 상기 복수의 컴퓨터 중 적어도 하나의 컴퓨터에게 송신하고,

상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터로부터 정보를 수신 -상기 정보는 상기 관리 팩에 기초하여 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터에 의해 수행된 구성 드리프트에 대한 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터의 평가에 기초한 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터의 구성 드리프트의 평가를 포함하고, 상기 구성 드리프트는 상기 적어도 하나의 컴퓨터의 실제 구성의 상기 구성 베이스라인으로부터의 일탈(deviation)을 포함함-

함으로써, 상기 복수의 컴퓨터의 구성 드리프트를 관리하게 하도록 구성된 모듈을 포함하는

시스템.

청구항 2

제 1 항에 있어서,

상기 모듈은 요청에 따라(on demand) 구성 드리프트를 평가하는

시스템.

청구항 3

제 1 항에 있어서,

상기 구성 드리프트는 상기 구성 규칙의 집합의 특정 구성 규칙에 대해 상기 복수의 컴퓨터 중 특정 컴퓨터에 대해서만 평가되는

시스템.

청구항 4

제 1 항에 있어서,

시간 기간에 걸쳐 서비스에 대해 수행되는 모든 교정이 유지되고 상기 서비스의 새로운 인스턴스에 적용됨으로써, 상기 구성 드리프트가 제거되는

시스템.

청구항 5

데이터센터 관리 컴퓨터 상에서 적어도 하나의 모듈을 실행함으로써 상기 데이터센터 관리 컴퓨터로부터 복수의 관리된 컴퓨터의 구성 드리프트를 관리하는 단계를 포함하되, 상기 적어도 하나의 모듈은:

구성 규칙의 집합을 정의함으로써 상기 복수의 관리된 컴퓨터에 대한 구성 베이스라인을 정의하고 -상기 구성 규칙의 집합의 각각은 프로그램, 운영체제, 서비스, 애플리케이션, 데이터베이스, 또는 웹사이트를 포함하는 관리된 엔티티를 포함하는 타겟과 연관되고, 상기 구성 베이스라인은 복수의 머신 각각으로부터 관독된 적어도 하나의 구성 설정을 결합함으로써 생성됨- ,

상기 구성 규칙의 집합을 포함하는 상기 정의된 구성 베이스라인에 기초하여 관리 팩(management pack)을 생성하며,

상기 관리 팩을 상기 복수의 컴퓨터 중 적어도 하나의 컴퓨터에게 송신하고,

상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터로부터 정보를 수신하며 -상기 정보는 상기 관리 팩에 기초하여 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터에 의해 수행된 구성 드리프트에 대한 상기 복수의 관리된 컴퓨터 중 상기 적어도 하나의 컴퓨터의 평가에 기초한 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터의 구성 드리프트의 평가를 포함하고, 상기 구성 드리프트는 상기 적어도 하나의 컴퓨터의 실제 구성의 상기 정의된 구성 베이스라인으로부터의 이탈(deviation)을 포함함- ,

상기 정의된 구성 베이스라인을 포함하는 상기 구성 규칙의 집합의 적어도 하나의 구성 규칙으로부터 상기 적어도 하나의 컴퓨터가 드리프트되었음을 판정하고,

상기 적어도 하나의 드리프트된 컴퓨터를 상기 구성 베이스라인에 합치하는 상태로 되돌리기 위해 구성 드리프트를 교정하는 명령어 세트를 이용하여, 상기 적어도 하나의 드리프트된 컴퓨터 상에서 상기 적어도 하나의 구성 규칙과 연관된 설정만을 요청에 따라(on demand) 교정하는

방법.

청구항 6

제 5 항에 있어서,

상기 구성 규칙의 집합 중 제1 구성 규칙과 제2 구성 규칙 사이에, 특정 설정이 상기 제1 구성 규칙과 상기 제2 구성 규칙 모두에 연관되는 의존성(dependency)이 존재한다는 판정에 응답하여, 상기 제1 구성 규칙과 상기 제2 구성 규칙 모두에 대해 검증 조건에 합치하는 상기 특정 설정에 대한 값을 결정하는 단계를 더 포함하는

방법.

청구항 7

제 6 항에 있어서,

상기 제1 구성 규칙과 상기 제2 구성 규칙 모두에 대해 상기 검증 조건에 합치하는 상기 특정 설정에 대한 제안된 값을 디스플레이하는 단계를 더 포함하는

방법.

청구항 8

제 5 항에 있어서,

상기 구성 베이스라인을 정의하는 것은, 제1 컴퓨터에 대해 제1 타겟과 연관된 상기 구성 베이스라인의 제1 구성 규칙을 식별하고, 제2의 다른 컴퓨터에 대해 제2의 다른 타겟과 연관된 상기 구성 베이스라인의 제2 구성 규칙을 식별함으로써 수행되는

방법.

청구항 9

제 8 항에 있어서,

상기 제1 컴퓨터에 대해 상기 구성 베이스라인의 제1 구성 규칙을 식별한 것에 응답하여, 제1 타겟에 대한 설정이 상기 제1 구성 규칙에 대한 상기 제1 컴퓨터의 상기 설정으로 설정되고, 상기 제2 컴퓨터에 대해 상기 구성 베이스라인의 제2 구성 규칙을 식별한 것에 응답하여, 제2 타겟에 대한 설정이 상기 제2 구성 규칙에 대한 상기 제2 컴퓨터의 상기 설정으로 설정되는

방법.

청구항 10

제 5 항에 있어서,

요청에 따라(on demand) 구성 드리프트에 대한 상기 적어도 하나의 컴퓨터의 상기 평가를 수행하는 단계를 더 포함하는

방법.

청구항 11

제 5 항에 있어서,

상기 구성 규칙의 집합 중 제1 구성 규칙의 설정의 값을 변경하는 단계와,

상기 구성 규칙의 집합 중 상기 제1 구성 규칙과 제2 구성 규칙 사이에 의존성이 존재한다는 판정에 응답하여, 자동으로 상기 제2 구성 규칙의 타겟을 재시작하는 단계 -상기 제 2 구성 규칙의 타겟은 프로그램, 운영체제, 서비스, 애플리케이션, 데이터베이스, 또는 웹사이트를 포함하는 관리된 엔티티를 포함함-

를 더 포함하는 방법.

청구항 12

실행될 경우에, 적어도 하나의 프로세서로 하여금,

구성 규칙의 집합을 정의함으로써 복수의 컴퓨터에 대한 베이스라인 구성을 정의하고 -상기 구성 규칙의 집합의 각각은 타겟과 연관되고, 상기 타겟은 관리된 엔티티의 레벨로 정의되며, 상기 구성 베이스라인은 다수의 기존 컴퓨터 각각으로부터 판독된 적어도 하나의 구성 설정을 결합함으로써 생성됨-,

상기 구성 규칙의 집합을 포함하는 상기 정의된 구성 베이스라인에 기초하여 관리 팩(management pack)을 생성하며,

상기 관리 팩을 상기 복수의 컴퓨터 중 적어도 하나의 컴퓨터에게 송신하고,

상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터로부터 정보를 수신하며 -상기 정보는 상기 관리 팩에 기초하여 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터에 의해 수행된 구성 드리프트에 대한 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터의 평가에 기초한 상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터의

구성 드리프트의 평가를 포함하고, 상기 구성 드리프트는 상기 적어도 하나의 컴퓨터의 실제 구성의 상기 구성 베이스라인으로부터의 일탈(deviation)을 포함함-,

상기 복수의 컴퓨터 중 적어도 하나의 컴퓨터가 상기 구성 베이스라인의 상기 구성 규칙의 집합 중 적어도 하나의 구성 규칙에 합치하지 않는다는(non in compliance) 판정에 응답하여, 상기 적어도 하나의 컴퓨터가 합치하지 않는 상기 적어도 하나의 구성 규칙에 대하여 상기 적어도 하나의 컴퓨터에 대한 다음 유지보수 윈도우(maintenance window)에 대해 상기 적어도 하나의 컴퓨터의 교정을 스케줄링 -상기 교정은 상기 적어도 하나의 컴퓨터를 상기 구성 베이스라인에 합치하는 상태로 되돌리기 위해 구성 드리프트를 교정하는 명령어 세트를 포함함- ,

함으로써 상기 복수의 컴퓨터의 구성 드리프트를 관리하게 하는 컴퓨터 실행가능 명령어를 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 13

제 12 항에 있어서,

실행될 경우에 상기 적어도 하나의 프로세서로 하여금,

상기 복수의 컴퓨터 중 상기 적어도 하나의 컴퓨터가 합치하지 않고 상기 적어도 하나의 구성 규칙에 대한 교정이 스케줄되었음을 보고하도록 하는

컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 14

제 13 항에 있어서,

실행될 경우에 상기 적어도 하나의 프로세서로 하여금,

서비스에 대한 기존 컴퓨터의 구성 베이스라인의 적어도 하나의 규칙을 판독하고 새로운 인스턴스에 대한 상기 서비스의 설정값을 상기 기존 컴퓨터의 설정값으로 설정함으로써 상기 서비스의 상기 새로운 인스턴스에 대한 베이스라인 구성을 정의하도록 하는

컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 15

제 13 항에 있어서,

실행될 경우에 상기 적어도 하나의 프로세서로 하여금,

상기 구성 베이스라인과 불합치하는 컴퓨터와 상기 구성 베이스라인에 합치하는 컴퓨터의 리스트를 디스플레이 하도록 하는

컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 16

제 15 항에 있어서,

실행될 경우에 상기 적어도 하나의 프로세서로 하여금,

상기 불합치하는 컴퓨터에 대한 타겟을 디스플레이하도록 하는

컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 17

제 12 항에 있어서,

실행될 경우에 상기 적어도 하나의 프로세서로 하여금,

서비스의 새로운 인스턴스를 생성 -시간 기간에 걸쳐 상기 서비스에 수행된 모든 교정을 유지하고, 상기 서비스의 상기 새로운 인스턴스에 상기 유지된 모든 교정을 적용함으로써 상기 새로운 인스턴스에 대한 구성 드리프트가 제거됨- 하도록 하는

컴퓨터 실행가능 명령어를 더 포함하는 컴퓨터 판독가능 저장 메모리.

청구항 18

제 1 항에 있어서,

상기 구성 베이스라인은 적어도 하나의 허용된 애플리케이션 및 적어도 하나의 금지된 애플리케이션에 관한 정보를 포함하는

시스템.

발명의 설명

배 경 기 술

[0001] 데이터 센터는 자주, 다양한 플랫폼에 걸쳐 배치되고, 심지어 지리적으로 격리된 위치에 배치될 수 있는 물리적 기계 및 가상 기계를 포함하는 복잡한 서버 환경을 관리한다. 데이터 센터는 수백 또는 심지어 수천의 서버를 포함할 수 있기 때문에, 데이터 센터 관리자는 그들이 서버 프로비저닝(provisioning) 및 업데이트를 자동화할 수 있게 하고, 다음 데이터 센터 확장을 계획할 수 있게 하며, 기타 데이터 센터 환경의 많은 도전을 처리하는 것을 도울 수 있게 하는 툴(tool)을 환영한다.

[0002] 스케줄되지 않은 시스템 다운타임 중 반 정도가 구성(configuration) 문제에 기인할 수 있다고 추정된다. 구성은, 시스템이 특정 목적을 위해 동작하도록 하드웨어 및 소프트웨어의 속성(attributes)에 대한 값을 식별하고 설정하는 프로세스를 지칭한다. 관리자가 매일 발생하는 문제 및 수요에 응답하면서, 서버는 점진적으로 정책 또는 표준에서 벗어나는 방식으로 구성되어질 수 있다. 이 프로세스는 구성 드리프트(configuration drift)라고 불린다.

발명의 내용

해결하려는 과제

[0003] 다수의 컴퓨터를 관리하는 것은 컴퓨터가 유사하게 구성되어 있지 않으면 급속히 복잡해질 수 있다. 처음에는 동일하게 구성된 컴퓨터라도, 시간에 따라 구성 변화가 발생할 수 있다. 구성 드리프트가 클수록 문제를 해결하는 것이 어려워지고, 컴퓨터의 효과적인 관리 및 유지보수를 더 어렵게 한다.

과제의 해결 수단

[0004] 컴퓨터에 대한 소프트웨어 구성 드리프트는, 희망 구성을 정의하고, 희망 구성으로부터의 컴퓨터의 일탈(구성 드리프트)을 산정하고 구성 드리프트를 제거하도록 컴퓨터를 (수동으로 또는 자동으로) 교정함으로써 관리될 수 있다. 희망 구성은 잘 구성된 것으로 생각되는 하나 이상의 기존 컴퓨터의 구성을 판독함으로써 정의될 수 있다. 구성 드리프트의 교정은 자동화되고 제어된 방식으로 수행될 수 있다. 배치 이미지/애셋(asset)과 배치된 시스템 사이의 드리프트를 방지하여 구성 드리프트 없이 이미지 기반 서비스, 재배포 및 스케일 아웃을 가능하게 하도록, 서비스 관련 배치(deployment) 베이스라인(baseline) 및 교정이 통합될 수 있다.

[0005] 이 개요는 아래에서 상세한 설명에 더 설명되는 개념 중 선택된 것을 단순화된 형태로 소개하기 위해 제공된다. 이 개요는 청구된 주제의 핵심 특징 또는 중요 특징을 식별하려는 것이 아니고, 청구된 주제의 범위를 제한하는데 사용하려는 것도 아니다.

도면의 간단한 설명

[0006] 도 1은 여기 개시된 주제의 태양에 따라 구성 드리프트를 관리하기 위한 시스템(100)의 예를 도시한다.
 도 2는 도 1에 대해 설명된 바와 같이 여기 개시된 주제의 태양에 따라 자동으로 구성 드리프트를 관리하기 위한 방법(200)의 예의 흐름도이다.
 도 3은 도 1에 대해 설명된 바와 같이 여기 개시된 주제의 태양에 따라 구성 드리프트를 관리하기 위해 생성될 수 있는 리포트의 일예이다.
 도 4는 도 2에 대해 설명된 바와 같이 여기 개시된 주제의 태양에 따라 구성 드리프트를 관리하기 위해 생성될 수 있는 다른 리포트의 일예이다.
 도 5는 여기 개시된 주제의 태양이 구현될 수 있는 연산 환경의 일예를 도시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0007] 개관

[0008] 시스템의 희망 구성을 관리하기 위해 모델 기반 접근이 사용될 수 있다. 구성 관리자는 구성 베이스라인을 정의함으로써 시스템의 구성 드리프트를 측정하고 관리할 수 있는 능력을 제공한다. 구성 베이스라인은 운영체제(operating system) 구성, 소프트웨어 업데이트, 금지 및 허용 애플리케이션 및 맞춤 설정에 관한 정보를 포함할 수 있다. 구성 베이스라인 정의는 데이터 센터의 관리자가 구성 드리프트를 관리할 수 있게 한다. 물리 및 가상 기계 컬렉션의 멤버에 변화를 적용할 시간을 조직이 선택할 수 있도록 유지보수 윈도우가 구현될 수 있다.

[0009] 여기 개시된 주제는, 구성 의도(intent)를 정의하고, 구성 드리프트를 산정하며 구성 드리프트를 교정함으로써 구성 드리프트를 관리하는 능력을 제공하기 위한 방법, 시스템 및 컴퓨터 프로그램 제품을 설명한다. 희망 구성은 구성 베이스라인을 생성함으로써 정의될 수 있다. 구성 베이스라인은 관련된 구성 규칙의 컬렉션을 포함할 수 있다. 예를 들어, 보안을 위한 구성 베이스라인은 보안 이슈 등에 관련된 구성 규칙을 포함할 수 있다. 구성 베이스라인은 컴퓨터 관리자 또는 소프트웨어 벤더(vendor) 또는 기타 사용자에게 의해 정의될 수 있다. 종종 컴퓨터 관리자는 그의 컴퓨터에 소프트웨어를 설치하고 다수의 설정 값을 조정함으로써 그 자신의 필요를 위해 소프트웨어를 조정한다. 소프트웨어 벤더는 관리자가 어떤 값을 변경할지 예측할 수 없기 때문에 소프트웨어 벤더에 의해 정의된 구성 베이스라인은 이런 경우에 유용하지 않을 수 있다. 새로운 구성 베이스라인을 정의하기 위해 변경된 설정을 추적하고 설정의 위치를 찾는 것은 번거로울 수 있다.

[0010] 여기 개시된 주제의 태양에 따라, 사용자는 하나 이상의 사전 정의된 구성 규칙을 포함하는 구성 베이스라인을 정의할 수 있다. 구성 규칙은, 관리되는 컴퓨터를 컴플라이언스(compliance)를 위해 체크하려는 사용자의 의도가 포착되도록 정의될 수 있다. 구성 규칙은 구성 설정과 연관될 수 있고 타겟 유형, 구성 설정에 대한 검증(validation) 및 교정을 정의할 수 있다.

[0011] 구성 설정은 여하한 관심 설정(예를 들어, 레지스트리 값의 이름, 버전 또는 수정일과 같은 파일 속성의 이름, 프로그램 요소 속성의 이름 등)일 수 있다. 타겟 유형은 설정이 영향을 주는 관리된 엔티티의 유형(예를 들어, SQL 서버, IIS 서버, 데이터베이스, 서비스, 웹사이트, 운영체제 등과 같은, 그러나 이에 제한되지 않는 프로그램 또는 애플리케이션)을 정의한다. 더 자세한(granular) 수준에서 타겟 유형을 정의하는 것(예를 들어, 컴퓨터 레벨과 같은 덜 자세한 수준이 아니라 SQL 서버 또는 IIS 서버 수준에서 타겟을 정의하는 것)은 많은 방식에서 도움이 된다. 예를 들어, 타겟 유형이 특정되므로, 특정 컴퓨터가 관리된 엔티티를 갖지 않는다면, 그 구성 규칙에 대해 컴퓨터를 평가할 필요가 없다. 또한, 구성 드리프트는 더 자세한 수준, 즉, 관리된 엔티티 수준에서 보고될 수 있다. 예를 들어, 구성 드리프트 리포트는, 예를 들어, 특정 컴퓨터 상에서 무엇인가가 잘못 구성되어 있다고 보고하는 것이 아니라, 이 특정 컴퓨터 상의 SQL 서버가 잘못 구성되어 있지만 IIS 서버는 그렇지 않다고 보고할 수 있다.

- [0012] 검증은 설정의 희망 값 또는 값의 범위(예를 들어, 2 초과, 11/23/2009와 같음 등)를 정의하여 특정 기계 또는 기계 그룹 상의 설정의 컴플라이언스 또는 넌컴플라이언스(noncompliance)가 판정될 수 있다. 교정은, 설정을 검증에서 정의된 바와 같은 그의 희망 값으로 만들거나 되돌리도록 실행되는 명령의 세트를 정의한다. 여기 개시된 주제의 태양에 따라, 구성 베이스라인은 기존 컴퓨터의 구성을 판독함으로써 생성될 수 있다. 예를 들어, 희망 방식으로 구성된 컴퓨터의 구성이 판독되고 다른 컴퓨터에 대한 구성 베이스라인을 생성하는데 사용될 수 있다. 소프트웨어 벤더 정의된 구성 베이스라인은, 소프트웨어의 새로운 인스턴스에 대한 구성 베이스라인을 생성하도록 사전에 구성된 컴퓨터를 판독하고 벤더 정의된 베이스라인을 적절하게 수정하는 소프트웨어에 의해 업데이트될 수 있다. 구성 베이스라인, 규칙 및 설정은 구성 데이터스토어에 저장될 수 있다.
- [0013] 일단 구성 베이스라인이 정의되면, 정의된 구성은 컴퓨터 그룹에 할당될 수 있다. 주기적으로, 스케줄에 따라, 또는 요청에 따라, 그룹 내의 각 컴퓨터 또는 그룹 내 컴퓨터의 부분집합(subset)은 그에 할당된 구성 베이스라인에 대해 평가될 수 있다. 평가가 컴퓨터가 동일 설정에 대해 구성 규칙에 특정된 값과 맞지 않는 설정에 대한 값을 갖는다고 판정하면, 컴퓨터는 희망 구성으로부터 드리프트된 것으로 판정된다. 여기 개시된 주제의 태양은 사용자가 하나 이상의 데이터센터 컴퓨터를 컴플라이언스를 위해 체크하고 사용자에게 컴플라이언스 드리프트를 경고할 수 있도록 한다.
- [0014] 컴퓨터의 구성이 그 컴퓨터에 대한 베이스라인 구성으로부터 이탈하는 때에, 구성 드리프트가 발생한 것이다. 컴퓨터가 구성 규칙에 대한 희망 구성으로부터 드리프트된 것으로 판정되는 때에, 컴퓨터 관리자는 구성 규칙에 대해 그 컴퓨터에 대한 교정 동작을 수행할 수 있다. 교정 동작은 구성 규칙에 대해 컴퓨터에 대한 구성 드리프트를 제거하기 위해, 교정을 위해 특정된 명령의 세트를 실행한다. 교정은 동일한 컴퓨터에 대해 다수의 구성 규칙에 대해 수행될 수 있다. 여기 개시된 주제의 태양에 따라, 교정은 제어된 방식으로 수행될 수 있다. 예를 들어, 관리자는 컴퓨터의 다음 서비스 윈도우에서 자동으로 교정을 수행하도록 선택할 수 있어서, 예를 들어, 컴퓨터의 사용자는 영향을 받지 않는다. 다르게는, 관리자는 요청에 따라 (실질적으로 즉시) 교정을 수행하라는 옵션을 제공받을 수 있다.
- [0015] 관리 팩이 구성 베이스라인으로부터 생성될 수 있다. 관리 팩은 컴플라이언스 정책의 생성 및 생성된 컴플라이언스 정책에 대한 관리되는 컴퓨터의 평가를 용이하게 하는 사용자-특정 의도 모델을 포함할 수 있다. 관리되는 컴퓨터 상의 정책 처리 엔진(policy processing engine)이 현재 상태(예를 들어, 설정의 실제 값)와 사용자-특정 의도 모델에 기초하여 컴플라이언스 리포트를 생성할 수 있고 의도된 또는 기대되는 값을 표시할 수 있도록 관리 팩은 관리되는 컴퓨터에 보내질 수 있다. 데이터센터 내 관리되는 컴퓨터는 스케줄된 방식으로 또는 요청에 따라 하나 이상의 정의된 구성 베이스라인에 대해 컴플라이언스 체크될 수 있다. 요청에 따른 스캔에 있어, 시스템은, 사용자가 사전 정의된 데이터 센터 내 특정한 관리되는 컴퓨터에 대해 구성 베이스라인 또는 구성 규칙을 선택적으로 실행할 수 있도록 한다. 구성 드리프트가 없는 경우에도, 시스템은 관심 설정의 의도된 값, 설정의 실제 값, 마지막 스캔 시간 및 스캔 자체의 상태(예를 들어, 스캔이 진행 중인지 여부 또는 그것이 실패 또는 성공했는지 여부)를 포함할 수 있는 사용자 스캔 정보를 제공한다.
- [0016] 컴플라이언스 상태는 분산 방식으로 계산될 수 있다. 즉, 데이터센터 관리 컴퓨터에서 컴플라이언스를 평가하는 대신에, 각각의 관리되는 컴퓨터는 구성 정책을 그 현재 상태에 대해 평가하는 컴플라이언스 정책 처리 엔진을 실행할 수 있다. 모든 교정 태스크, 구성 드리프트로부터 복원하는 태스크는 데이터센터 관리 컴퓨터 상에서 스케줄될 수 있다. 교정 태스크 상태(스케줄됨, 진행중, 완료 또는 실패)는 구성 규칙 상에서 상태로서 설정될 수 있다.
- [0017] 구성 베이스라인은 특정 인스턴스와 연관되거나 서비스 템플릿 내 기계 층(machine tier) 템플릿의 모든 인스턴스와 연관될 수 있다. 이는 템플릿 배치의 일부로서 배치되고 있는 새로이 프로비저닝되는 기계 각각에 대해 또는 기계 층의 스케일 아웃 동안 검증 및 교정을 포함하는 구성 규칙의 적용을 가능하게 한다. 교정은 스케줄되거나 요청에 따라 실행될 수 있다. 가이드된 컴플라이언스 행동은 3 단계(phase)로 수행될 수 있다. 3 단계는 설정을 스캔하는 단계, 컴플라이언스 스캔을 수행하는 단계 및 컴플라이언스 정보를 보고하는 단계이다. 3 단계 각각은 독립적으로 모델링될 수 있어서 상이한 관심 설정이 각각의 단계에 대해 사용될 수 있다.
- [0018] **구성 드리프트 관리**
- [0019] 도 1은 여기 개시된 주제의 태양에 따라 구성 드리프트를 관리하기 위한 시스템(100)의 일예를 도시한다. 도 5에 대해 후술되는 컴퓨터와 같은 하나 이상의 컴퓨터 상에 시스템(100)의 전부 또는 일부가 상주할 수 있다.

다르게는, 시스템(100) 또는 그 부분은 독립(stand-alone) 시스템으로서 또는 플러그인(plug-in)이나 애드인(add-in)으로서 제공될 수 있다. 시스템(100)의 전부 또는 부분은 데이터센터 시스템 센터 컴퓨터 상에 상주할 수 있다. 시스템(100)의 전부 또는 부분은 관리되는 컴퓨터 상에 상주할 수 있다.

[0020] 시스템(100)에 의해 관리되는 컴퓨터는 하나 이상의 물리적 기계, 하나 이상의 가상 기계 또는 물리적 기계와 가상 기계의 여하한 결합을 포함할 수 있다. 도 1에서, 관리되는 컴퓨터(140)는 물리적 기계(122)를 나타낸다. 관리되는 컴퓨터(134)는 가상 기계 호스트(120)를 나타내고 관리되는 호스트 컴퓨터(132)는 2개의 가상 게스트 기계, 게스트(136)와 게스트(138)를 호스팅한다. 가상 기계(virtual machine; VM)는 물리적 기계와 같이 프로그램을 실행하는 컴퓨터의 소프트웨어 구현이다. 데이터 센터 가상화(virtualization)에서, 추상 계층(abstract layer)은 애플리케이션과 기저(underlying) 서버, 스토리지 및 네트워크 인프라스트럭처 사이에서 생성된다. 가상 기계는 상이한 가상 기계 사이에서 기저의 물리적 기계 자원이 공유되도록 허용할 수 있다. 가상화는 다수의 물리적 엔티티로부터 하나의 논리적 엔티티를 생성할 수 있고, 또는 연산, 스토리지, 네트워크 또는 애플리케이션 자원을 포함하는 하나의 물리적 엔티티로부터 많은 논리적 엔티티를 생성할 수 있다.

[0021] 각각의 가상 기계는 그 자신의 운영 체제를 실행할 수 있다. 가상화를 제공하는 소프트웨어 계층은 가상 기계 모니터 또는 하이퍼바이저라고 불린다. 하이퍼바이저는 하드웨어 상에서 또는 운영 체제 상부에서 직접 실행될 수 있다. 가상 기계 환경에서, 다수의 운영 체제 환경은 동일한 컴퓨터에서 공존할 수 있지만, 서로로부터 격리를 유지할 수 있다. 가상 기계는, 다른 물리적 호스트 컴퓨터 시스템의 한계 내에서 추가의 별도 컴퓨터 시스템으로서 동작하는 전자 파일의 세트를 포함할 수 있다.

[0022] 시스템(100)은 컴퓨터(102)와 같은 하나 이상의 컴퓨터를 포함할 수 있다. 하나 이상의 컴퓨터는, 프로세서(프로세서(142) 등), 메모리(144)와 같은 메모리, 하나 이상의 컴퓨터 상에서 구성 드리프트를 관리하기 위한 하나 이상의 모듈 중 하나 이상을 포함할 수 있다. 다른 공지된 컴포넌트도 포함될 수 있지만 여기에 보여지지 않는다. 구성 드리프트를 관리하기 위한 모듈(들)은 메모리(144)에 로딩될 수 있어서 프로세서(142)와 같은 하나 이상의 프로세서가 구성 드리프트를 관리하기 위한 모듈(들)에 속하는 동작을 수행하도록 함을 인식할 것이다. 구성 드리프트를 관리하기 위한 모듈(들)은 구성 관리자(106)와 같은 구성 관리자를 포함할 수 있다. 구성 관리자(106)는 교정 서비스(110)와 같은 교정 서비스와 구성 서비스(108)와 같은 구성 서비스를 포함할 수 있다.

[0023] 구성 관리자(106)와 교정 서비스(110)는 상술한 바와 같이 다수의 컴퓨터, 물리적 또는 가상, 호스트 또는 게스트와 통신할 수 있다. 여기에 개시된 주제의 태양에 따라, 관리되는 컴퓨터 또는 컴퓨터 그룹에 대한 구성 정보는 컴퓨터(102)와 같은 데이터센터 시스템 센터 컴퓨터 상에서 입력될 수 있다. 구성 베이스라인이 생성될 수 있고 구성 베이스라인이 타겟 컴퓨터 또는 타겟 컴퓨터 (관리되는 컴퓨터)의 그룹에 할당될 수 있다. 구성 베이스라인은 구성 규칙(112)의 세트 등을 생성함으로써 생성될 수 있다. 구성 베이스라인은 데이터스토어(104)와 같은 구성 데이터 스토어에 저장될 수 있다.

[0024] 구성 규칙(112)과 같은 구성 규칙은, 구성 설정(112a)과 같은 특정 설정을 선택하고, 설정에 대한 값, 값의 목록 또는 값의 범위를 제공함으로써 생성될 수 있다. 할당된 값이 수용가능한 값 또는 값의 목록/범위에 맞는지 여부를 결정할 수 있는, 검증(112b)과 같은 검증 알고리즘이 생성될 수 있다. 구성 규칙은 타겟(112c)과 같은 타겟 유형의 상세(specification)을 포함할 수도 있다. 예를 들어, 타겟 유형은, 프로그램, 애플리케이션, 서비스, 데이터베이스, 웹사이트, 운영체제 등을 포함하지만 이에 제한되지 않는 여하한 관리된 엔티티일 수 있다. 타겟 시스템 또는 타겟 컴퓨터의 그룹이 특정될 수 있다. 구성 규칙은 설정을 희망 값으로 되돌리기 위해 취해져야할 동작 또는 동작 세트를 포함할 수 있다. 하나 이상의 설정을 희망 값 또는 값들로 되돌리는 프로세서는 교정이라고 불린다. 도 1에서 교정(112d)은 이들 동작을 나타낸다.

[0025] 사용자에게 의해, 또는 특정한 관리되는 컴퓨터로부터, 예를 들어, 관리되는 컴퓨터(134) 또는 관리되는 컴퓨터(132) 또는 관리되는 컴퓨터(140) 등으로부터 설정에 대한 값을 얻음으로써 설정에 대한 희망 값이 설정에 할당될 수 있다. 맞지 않는(noncompliant) 값이 어떻게 통신될지 및 누구에게 경고가 전달되는지를 설명하는 경고 규칙도 특정될 수 있다. 112e로 나타내어진 다른 정보도 구성 규칙에 추가될 수 있다. 구성 규칙(112) 등과 같은 구성 규칙의 세트가 생성될 수 있고 구성 베이스라인(114)과 같은 하나 이상의 구성 베이스라인으로 수집될 수도 있다. 구성 베이스라인은 타겟 컴퓨터(116)와 같은 하나 이상의 타겟 컴퓨터에 할당될 수 있다. 구성 베이스라인, 규칙 및 타겟 그룹은 구성 데이터스토어 또는 데이터베이스(예를 들어, 데이터스토어(104))에 저장될 수 있다.

[0026] 상술한 정보는 수집되어 관리 팩(118)과 같은 관리 팩을 생성하는데 사용될 수 있다. 관리 팩은 정보가 전송되는 위치를 식별할 수 있고 관리되는 컴퓨터가 합치하는지(compliant) 여부를 판정하게될 테스트를 어떻게 실행

하는지 설명할 수 있다. 구성 서비스(108)와 같은 구성 서비스는 그 후 관리 팩(예를 들어, 관리 팩(118))을 구성 베이스라인의 타겟에서 식별되는 컴퓨터에게 배포할 수 있다.

[0027] 관리되는 컴퓨터(134) 등의 관리되는 컴퓨터 상의 헬스 서비스(health service)(126)와 같은 헬스 서비스는 관리 팩에서 설명되는 구성에 대해 태스크를 실행하도록 동작할 수 있다. 관리 팩에서 설명되는 태스크에는 컴플라이언스 정책의 생성이 포함될 수 있다. 컴플라이언스 정책 프로세서(128)와 같은 컴플라이언스 정책 프로세서는 관리되는 컴퓨터(예를 들어, 관리되는 컴퓨터(134))가 하나 이상의 구성 베이스라인에 맞는지 여부를 판정할 수 있다. 도 3 및 4에 대해 도시되고 아래에서 자세히 설명되는 리포트와 같은, 그러나 이에 제한되지 않는 리포트가 생성될 수 있다. 리포트는 모니터 또는 기타 디스플레이 장치에 표시되거나 인쇄될 수 있다. 리포트는 하나 이상의 설정이 설정될 수 있는 제안 값에 대한 정보를 포함할 수 있다. 몇몇의 프로그램은 동일한 설정 또는 설정들을 이용할 수 있음을 인식할 것이다. 이 경우, 제안 값이 하나 이상의 애플리케이션의 요구사항에 합치하도록 규칙이 결합될 수 있다. 예를 들어, 한 애플리케이션은 특정 설정에 대한 값이 3보다 큰을 특정할 수 있는 한편, 제2의 애플리케이션은 그 설정에 대한 값이 7 미만임을 특정할 수 있다. 컴플라이언스 리포트는 설정에 대한 값이 4, 5 또는 6일 것을 제한할 수 있다. 또한, 컴플라이언스 리포트는 5와 같은 특정 선택을 제시할 수 있다.

[0028] 구성 드리프트 정보를 포함하는 컴플라이언스 결과가 데이터센터 시스템 센터 컴퓨터에 전송되거나 그에 의해 불러오기(retrieve)될 수 있고, 컴플라이언스 데이터스토어(124)와 같은 컴플라이언스 데이터스토어에 저장될 수 있다. 교정은 즉시(요청에 따라) 또는 다음 유지보수 윈도우 동안 등 미래의 시간에 스케줄되어 자동으로 수행될 수 있다. 대신에, 사용자는 교정에 대한 필요성을 통지받을 수 있고 교정이 수동으로 수행될 수 있다. 컴퓨터의 컴플라이언스 상태 및 사용자에 의해 취해진 행동을 모두 포함하는 정보가 사용자에게 제시될 수 있다. 예를 들어, 특정 컴퓨터 상의 애플리케이션의 특정 인스턴스가 특정 규칙에 대해 합치하지 않고 개선이 다음 유지보수 윈도우에 대해 스케줄되었음 또는 아직 스케줄되지 않았음을 나타내는 정보가 사용자에게 제시될 수 있다.

[0029] 일 기간 동안 이루어진 모든 교정은 교정 데이터스토어(105)와 같은 교정 데이터스토어에 유지 또는 저장될 수 있어서, 다른 유사한 시스템 또는 인스턴스가 추가되면 유지된 교정의 전부가 새로운 시스템에 적용될 수 있다. 예를 들어, 기계가 미리 생성된 운영체제 이미지를 이용하여 셋업되었고, 후속하여 기계에 교정이 적용되었다고 가정하자. 동일한 이미지가 다른 시스템을 배치하는데 사용되는 경우, 교정에 의해 이루어진 변화는 없어질 것이다. 여기 개시된 주제의 태양에 따라, 이루어진 모든 교정이 유지된다. 운영체제 이미지가 다른 시스템을 배치하는데 사용되는 때에, 유지되는 교정이 새로운 시스템에 적용되어 구성 드리프트를 제거한다.

[0030] 도 3은 산출될 수 있는 컴플라이언스 리포트의 일예이다. 도 3은, 뉴욕에 하나(NEW YORK(304))와 샌 프란시스코에 하나(SAN FRAN(306))의 2개의 영업소(site)를 갖는 회사 Woodgrove IT (WG IT(302))에 대한 컴플라이언스 리포트를 나타낸다. 뉴욕 영업소는 세 개의 기계 그룹, 개발 그룹(DEV(308)), 생산 그룹(PROD(310)) 및 스테이징 그룹(STAG(312))을 갖는다. 예에서, 사용자는 리포트를 수신하기 위해 생산 기계 그룹을 선택하였다(SEARCH PRODUCTION(314)이라는 제목에 반영됨). 리포트의 바디(316)에 열거된 기계는 뉴욕 영업소의 생산 기계를 나타낸다. 리포트로부터, 식별자 CI NewYorkProd(318)를 갖는 컴퓨터는 합치(compliant)하는 반면(320), 식별자 Computer2_NYP2(322)를 갖는 컴퓨터는 합치하지 않는다(noncompliant)(324)는 것을 볼 수 있다. 또한, 컴플라이언트 리포트(300)는 어떤 종류의 베이스라인이 합치하지 않는지 보여준다.

[0031] 도 3에서, 패치 베이스라인(Patch Baseline)(326)은 합치하지 않는다(328). 합치하지 않는다(322) Computer5_NYP2(330)를 제외하고는, 리포트에 열거된 모든 다른 컴퓨터는 합치한다. Computer5_NYP2(330)를 선택하는 것은 Computer5_NYP2(330)에 대한 합치 및 불합치 애플리케이션 베이스라인의 목록을 가져올 수 있다. 그러므로, 이 리포트는 주어진 컴퓨터 또는 컴퓨터 그룹의 그 컴퓨터와 연관된 구성 베이스라인에 대한 합치 또는 불합치의 상태에 관한 정보를 제공할 수 있다. 특정 컴퓨터 또는 컴퓨터 그룹이 불합치하면, 리포트는 어떤 기계가 합치하고 어떤 기계가 불합치하며 그 기계가 구성 베이스라인 내 어떤 특정 규칙 또는 규칙들에 대해 불합치한지를 식별하는 정보를 제공할 수 있다. 제시된 다른 정보는 마지막 스캔의 일시(334) 및 다음 스케줄된 스캔의 일시(336)를 포함하지만 이에 제한되지 않는다.

[0032] 도 2는 여기에 개시된 주제에 따라 구성 드리프트를 관리하기 위한 방법(200)의 일예를 도시한다. 설명된 방법 또는 그 부분은 도 1과 관련하여 상술한 바와 같은 시스템(100) 또는 그 부분에 의해 수행될 수 있다. 202에서 설계 의도가 정의될 수 있다.

[0033] 컴퓨터를 구성하는 것은 단순히 설정을 선택하고 설정에 값을 할당하는 것 이상이다. 이는 도전적인 과업일 수

있다. 먼저, 관련된 설정의 갯수 자체가 도전이다. 여하한 컴퓨터 상에서도, 매우 많은 수의 가능한 설정이 있다. 가능한 설정의 수는 사용자가 관리할 수 있는 것을 훨씬 넘을 수 있고 자동화된 시스템이 검증할 수 있는 것 이상일 수도 있다. 그들 설정 각각에는 다수의 값이 할당될 수 있다. 개시된 주제의 태양에 따라, 특정 세트의 설정이 모니터링을 위해 식별된다.

[0034] 다음으로, 애플리케이션 제공자나 사용자나 그 자신이 효율적으로 구성 베이스라인을 생성할 수 없을 수 있다. 애플리케이션 벤더는 어떤 설정이 그의 애플리케이션에 영향을 주는지 알 수는 있지만, 어떤 설정 값이 사용자의 환경에서 가장 효율적으로 동작할지는 알지 못할 수 있다. 예를 들어, SQL 애플리케이션의 제공자는 애플리케이션의 기능이 특정 레지스트리 키에 의존한다는 것을 알 수 있지만, 사용자의 환경에서 어떤 값이 레지스트리 키에 설정될 것인지는 알지 못할 수 있다. 유사하게, 사용자는 그의 시스템에서 실행되는 애플리케이션에 어떤 설정이 영향을 주는지 알지 못할 수 있지만 사용자의 환경에서 애플리케이션이 잘 수행되게 하는 값으로 특정 설정의 값을 설정함으로써 시스템을 조율하였을 수도 있다. 여기 개시된 주제의 태양에 따라, 어떤 설정이 관련되거나 설정에 대한 값이 무엇인지 알지 못할 수 있는 사용자는 희망되는 방법으로 구성된 기계를 식별할 수 있다. 자동화된 특성은 잘 동작하는 기계의 설정을 관독할 수 있고 식별된 기계로부터의 설정의 값이 베이스라인 구성 또는 그 일부를 생성하는데 사용될 수 있다. 식별된 기계로부터 관독된 값 및 설정은 운영체제 설정 또는 애플리케이션 수준 설정을 포함할 수 있다. 생성된 베이스라인 구성은 서비스의 새로운 인스턴스 또는 새로운 기계의 컴플라이언스를 평가하는데 사용될 수 있다.

[0035] 마지막으로, 상술한 도전에 추가하여, 또 다른 도전은 설정 사이에 의존 또는 다수의 의존이 존재할 수 있다는 것이다. 즉, 여하한 주어진 설정이 하나 이상의 프로그램 또는 애플리케이션에 의해 사용될 수 있다. 그러므로, 하나의 설정의 값을 변경하는 과정에서 사용자는 다른 프로그램 또는 애플리케이션의 실행에 부정적으로 영향을 줄 수 있다. 예를 들어, 어떤 설정이 애플리케이션 A와 애플리케이션 B 모두에 의해 사용된다고 가정하자. 사용자가 애플리케이션 A에 대한 설정을 변화시키면, 애플리케이션 B가 그 실행 동안 변화되지 않는다고 기대하는 값이 기대치않게 변화되었으므로 애플리케이션 B가 재시작되어야 할 수 있다. 이는 애플리케이션 사이의 의존이 어떻게 구성 관리에 영향을 미치는지의 하나의 예일 뿐임을 인식할 것이다. 그러므로, 여기 개시된 주제의 태양에 따라, 자동화된 특성은 설정 값 변화에 의해 영향을 받는 애플리케이션을 결정할 수 있고 여하한 영향받는 프로그램 또는 애플리케이션을 자동으로 재시작할 수 있다.

[0036] 또한, 하나 이상의 프로그램이 특정 설정에 의존하고 제1 프로그램은 제1 범위에 있는 설정 값에 의존하며 제2 프로그램은 제2 범위에 있는 설정 값에 의존하는 경우, 자동화된 특성은 제1 및 제2 프로그램 모두와 함께 작동하는 값을 결정하기 위해 규칙을 결합할 수 있다. 예를 들어, 자동화된 특성은 제1 및 제2 범위 양자 모두 내에 있는 설정에 대한 값을 결정할 수 있다. 상술한 예에서 모두가 특정 설정을 이용하는 2개의 프로그램이 존재하지만, 자동화된 특성은 2개 애플리케이션에 제한되지 않음을 이해할 것이다: 애플리케이션 전부 또는 대부분과 작동하는 값을 결정하기 위해 여하한 수의 애플리케이션 값 범위가 산출될 수 있다. 설정에 의해 영향받는 모든 프로그램에 대해 검증하는 검증이 생성될 수 있다. 자동화된 특성은 여하한 영향받는 구성 규칙을 조정할 수 있다.

[0037] 다르게는, 구성된 설정이 선택될 수 있고 설정에 대해 값이 설정될 수 있다. 벤더의 모델 설정을 이용함으로써, 또는 벤더의 모델 설정으로 시작하고 다른 기계에 대한 설정을 관독하여 그들을 변경하며 벤더의 모델 설정을 맞추어 변경함으로써 구성 설정이 정의될 수 있다. 상술한 바와 같이, 구성 규칙은 구성 설정, 검증, 타겟 유형 및 구성 드리프트를 교정하는 동작의 세트(교정)를 포함할 수 있다. 여기 개시된 주제의 태양에 따라, 각 규칙에 대한 타겟 유형은 애플리케이션, 프로그램 또는 운영 체제 수준의 입도(granularity)에서와 같은 관리된 엔티티의 수준에서 정의될 수 있다. 예를 들어, 사용자는 특정 규칙과 연관된 타겟이 SQL 서버, Exchange 서버 또는 애플리케이션 A 등과 같은 그러나 이에 제한되지 않는 특정 애플리케이션임 특정하거나, IIS 웹사이트와 같은 그러나 이에 제한되지 않는 웹사이트임을 특정하거나, SQL 데이터베이스와 같은 그러나 이에 제한되지 않는 데이터베이스임을 특정하거나 할 수 있다. 이 입도 수준에서 정의된 타겟은 상술한 바와 같이 더 유익한(informative) 보고를 가능하게 하고 또한 아래에서 더 설정하는 바와 같이 더 잘 목표에 지향된 평가를 가능하게 한다.

[0038] 204에서, 기계 또는 기계의 세트는 그 기계 또는 기계 세트에 대한 구성 베이스라인과의 합치에 대해 평가될 수 있다. 사용자는 어떤 기계 또는 기계의 그룹이 평가되는지 제어할 수 있다. 사용자는 기계 또는 기계 그룹이 어떤 구성 규칙에 대해 평가되는지 제어할 수 있다. 평가의 스케줄링이 제어될 수 있고, 또는 다르게는, 평가는 요청에 따라 즉시 수행될 수 있다. 타겟이 관리된 엔티티의 수준에서 정의되기 때문에, 타겟 애플리케이션 또는 프로그램을 갖는 기계만이 평가되므로 평가 프로세스가 간소해진다. 예를 들어, 기계 A에 대한 구성 베이스

스라인에서 구성 규칙이 특정 설정에 대해 이메일 애플리케이션의 특정 유형의 타겟 유형만을 특정하고 기계 A는 이메일 애플리케이션의 그 특정 유형을 갖지 않는다면, 설정은 그 기계 상에서 평가될 필요가 없다.

[0039] 206에서, 평가에 관한 정보가 사용자에게 제시될 수 있다. 정보는 인쇄된 리포트 또는 디스플레이 등의 형태로 제시될 수 있다. 타겟이 관리된 엔티티의 수준에서 정의되기 때문에, 사용자에게 제시되는 정보는 더 유익할 수 있다. 예를 들어, 특정 기계가 구성 베이스라인에 불합치한다고 단순히 보고하는 대신에, 사용자에게 제시되는 정보는 도 3과 관련하여 상술한 바와 같이 불합치하는 특정 애플리케이션 또는 프로그램을 포함할 수도 있다.

[0040] 208에서, 교정이 수행될 수 있다. 교정은, 교정된 기계가 구성 베이스라인에 합치하는 상태로 돌아오도록 구성 드리프트를 정정하기 위해 수행되는 동작의 세트를 지칭한다. 교정은 수동으로 수행될 수 있거나, 자동으로 수행될 수 있다. 교정은 하나 이상의 컴퓨터 상에서(예를 들어, 선택된 컴퓨터 또는 컴퓨터 그룹 상에서) 수행될 수 있다. 교정은 하나 이상의 컴퓨터 상에서 하나 이상의 선택된 구성 규칙에 연관된 설정만으로 제한될 수 있다. 기계가 드리프트 된 것으로 판정되었다면, 사용자는 교정이 자동으로 수행되도록 선택할 수 있거나, 사용자는 교정을 수동으로 수행하도록 선택할 수 있다. 사용자가 자동 교정을 선택하면, 자동 교정이 즉시 수행될 수 있거나 기계의 다음 유지보수 윈도우에서 수행될 수 있다. 예를 들어, 기계가 다수의 사용자에게 서비스하는 경우, 유지보수 윈도우에서 교정을 수행하는 것은 사용자가 기계를 사용하고자 하는 때에 시스템이 동작하는 것을 보장할 수 있다.

[0041] 210에서, 애플리케이션, 프로그램, 애플리케이션의 그룹, 운영체제 또는 기계의 새로운 인스턴스가 구성 드리프트 없이 기존의 운영체제 이미지로부터 생성될 수 있다. 여기 개시된 주제의 태양에 따라, 적용된 여하한 교정이 유지되고 새로운 인스턴스가 생성되는 때에 사용된다.

[0042] 도 4는 생성될 수 있는 리포트(400)의 일예이다. 리포트(400)는 선택된 컴퓨터, 즉 선택된 컴퓨터(402)의 구성을 판독한 결과를 표시한다. 각각의 규칙(예를 들어 규칙 1(404) 내지 규칙 4(406))에 대해, 리포트는 상태(예를 들어, 규칙이 성공적으로 판독되었음(408) 또는 규칙이 판독 준비되어 있음(410)) 및 검증 규칙(예를 들어, 동일 5(412))를 나타낼 수 있다. 도 4에서, 선택된 컴퓨터(402)는 새로운 인스턴스의 생성을 위한 모델인 컴퓨터를 나타낼 수 있다. 자동화된 특성은 선택된 컴퓨터에 접촉할 수 있고 구성 베이스라인에 정의된 설정 각각에 대한 값을 판독할 수 있다. 이 구성 베이스라인은 회망에 따라 다른 서버에 적용될 수 있다. 검증 규칙 열(규칙 1(404)에 대한 동일 5(412))은 선택된 컴퓨터(402) 상의 해당하는 규칙에 대한 구성 설정의 값을 나타낸다. 각각의 규칙은 상이한 컴퓨터로 지시(point)될 수 있다. 예를 들어, 구성 베이스라인 내 각각의 규칙에 대해, 규칙 1은 컴퓨터 1을 지시하도록 설정될 수 있고, 규칙 2는 컴퓨터 2를 지시하도록 설정할 수 있는 등이다. 응답하여, 구성 설정이 규칙 1에 대한 구성 설정에 대해 컴퓨터 1로부터 판독될 것이고, 규칙 2에 대한 구성 설정에 대해 컴퓨터 2로부터 판독될 것이다. 이는 사용자가 다수의 기계로부터 구성 베이스라인의 일부를 생성할 수 있도록 한다. 즉, 구성 베이스라인은 SQL 애플리케이션과 같은 제1 타겟에 대해 잘 구성된 하나의 기계의 조합으로부터, 운영체제와 같은 제2 타겟에 대해 잘 구성된 제2 기계로부터, 다른 애플리케이션에 대해 잘 구성된 제3 기계로부터, 등으로 생성될 수 있다.

[0043] 적당한 연산 환경의 예

[0044] 여기에 개시된 주제의 다양한 태양에 대한 맥락을 제공하기 위해, 도 5 및 다음의 논의는 다양한 실시형태가 구현될 수 있는 적당한 연산 환경(510)의 간략한 일반적 설명을 제공하려는 것이다. 여기 개시된 주제가, 하나 이상의 컴퓨터 또는 기타 연산 장치에 의해 실행되는 프로그램 모듈과 같은 컴퓨터 실행가능 명령의 일반적인 맥락에서 설명되지만, 당업자는 여기 개시된 주제의 일부가 다른 프로그램 모듈 및/또는 하드웨어와 소프트웨어의 결합과 결합하여 구현될 수도 있음을 인식할 것이다. 일반적으로, 프로그램 모듈은, 특정 태스크를 수행하거나 또는 특정 데이터 유형을 구현하는 루틴(routine), 프로그램, 객체(object), 물품(physical artifact), 데이터 구조 등을 포함한다. 통상적으로, 프로그램 모듈의 기능은 다양한 실시형태에서 회망에 따라 결합되거나 분산될 수 있다. 연산 환경(510)은 적당한 동작 환경의 단지 하나의 예일 뿐이고, 여기 개시된 주제의 용도 또는 기능의 범위를 제한하려는 것이 아니다.

[0045] 도 5를 참조하여, 컴퓨터(512) 형태의 선형 스택(linear stack) 상에서 공통 루틴(courtine)의 효과적인 재시작을 위한 연산 장치가 설명된다. 컴퓨터(512)는 처리 유닛(514), 시스템 메모리(516) 및 시스템 버스(518)를 포함할 수 있다. 처리 유닛(514)은 다양한 가용 프로세서 중 여하한 것일 수 있다. 듀얼 마이크로프로세서 및

기타 멀티프로세서 아키텍처도 프로세서 유닛(514)으로 채용될 수 있다. 시스템 메모리(516)는 휘발성 메모리(520)와 비휘발성 메모리(522)를 포함할 수 있다. 비휘발성 메모리(522)는 ROM(read only memory), PROM(programmable ROM), EPROM(electrically programmable ROM) 또는 플래시 메모리를 포함할 수 있다. 휘발성 메모리(520)는 외부 캐시 메모리로서 동작할 수 있는 RAM(random access memory)을 포함할 수 있다. 시스템 버스(518)는 시스템 메모리(516)를 포함하는 시스템 물품(physical artifacts)을 처리 유닛(514)에 결합한다. 시스템 버스(518)는, 메모리 버스, 메모리 제어기, 주변 버스, 외부 버스 또는 로컬 버스를 포함하는 여러 유형 중 여하한 것일 수 있고, 다양한 가용 버스 아키텍처 중 여하한 것을 사용할 수 있다.

[0046] 컴퓨터(512)는 통상, 휘발성 및 비휘발성 매체, 제거가능(removable) 및 비제거가능 매체와 같은 다양한 컴퓨터 판독가능 매체를 포함한다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 여하한 방법 또는 기술로 구현될 수 있다. 컴퓨터 저장 매체는, RAM, ROM, EEPROM, 플래시 메모리 또는 기타 메모리 기술, CDROM, DVD(digital versatile disk) 또는 기타 광 디스크 저장소, 자기 카세트, 자기 테이프, 자기 디스크 저장소 또는 기타 자기 저장 장치, 또는 희망 정보를 저장하는데 사용될 수 있고 컴퓨터(512)에 의해 액세스될 수 있는 여하한 기타 비-일시적(non-transitory) 매체를 포함하지만 이에 제한되지 않는다.

[0047] 도 5는 사용자와 컴퓨터 리소스 사이의 중간자로서 동작할 수 있는 소프트웨어를 설명함을 인식할 것이다. 이 소프트웨어는 디스크 저장소(524) 상에 저장될 수 있고, 컴퓨터 시스템(512)의 리소스를 제어하고 할당할 수 있는 운영체제(528)를 포함할 수 있다. 디스크 저장소(525)는 인터페이스(526)와 같은 비제거가능 메모리 인터페이스를 통해 시스템 버스(518)로 접속된 하드 디스크 드라이브일 수 있다. 시스템 애플리케이션(530)은, 시스템 메모리(516) 또는 디스크 저장소(524)에 저장된 프로그램 모듈(532) 및 프로그램 데이터(534)를 통해 운영체제(528)에 의한 리소스 관리를 이용한다. 컴퓨터는 다양한 운영체제 또는 운영체제의 조합으로 구현될 수 있음을 인식할 것이다.

[0048] 사용자는 입력 장치(들)(536)을 통해 컴퓨터(512)로 명령 또는 정보를 입력할 수 있다. 입력 장치(536)는 마우스, 트랙볼, 스타일러스 터치패드, 키보드, 마이크로폰 등과 같은 포인팅 장치를 포함하지만 이에 제한되지 않는다. 이들 및 기타 입력 장치는 인터페이스 포트(들)(538)을 통해 시스템 버스(518)를 거쳐 처리 유닛(514)에 접속된다. 인터페이스 포트(들)(538)은 직렬 포트, 병렬 포트, USB(universal serial bus) 등을 나타낼 수 있다. 출력 장치(들)(540)은 입력 장치와 동일한 유형의 포트를 이용할 수 있다. 출력 어댑터(542)는 특정 어댑터를 요구하는 모니터, 스피커 및 프린터 등의 몇몇 출력 장치(540)가 존재함을 나타내기 위해 제공된다. 출력 어댑터(542)는 출력 장치(540)와 시스템 버스(518) 사이의 접속을 제공하는 비디오 및 사운드 카드를 포함하지만 이에 제한되지 않는다. 원격 컴퓨터(들)(544)과 같은 다른 장치 및/또는 시스템 및/또는 장치는 입력 및 출력 기능을 모두 제공할 수 있다.

[0049] 컴퓨터(512)는 원격 컴퓨터(들)(544)과 같은 하나 이상의 원격 컴퓨터로의 논리적 접속을 이용하여 네트워크된 환경에서 동작할 수 있다. 원격 컴퓨터(544)는 퍼스널 컴퓨터, 서버, 라우터, 네트워크 PC, 피어 장치(peer device) 또는 기타 공통 네트워크 노드일 수 있고, 통상적으로 컴퓨터(512)에 대해 상술한 요소 중 많은 것 또는 전부를 포함하는데, 도 4에는 메모리 저장 장치(546)만이 도시되었다. 원격 컴퓨터(들)(544)은 통신 접속(550)을 통해 논리적으로 접속될 수 있다. 네트워크 인터페이스(548)는 LAN(local area networks) 및 WAN(wide area networks)와 같은 통신 네트워크를 아우르지만, 다른 네트워크도 포함할 수 있다. 통신 접속(들)(550)은 네트워크 인터페이스(548)를 버스(518)에 접속하는데 채용되는 하드웨어/소프트웨어를 지칭한다. 접속(550)은 컴퓨터(512) 내부 또는 외부에 있을 수 있고, 모뎀(전화, 케이블, DSL 및 무선) 및 ISDN 어댑터, 이더넷(Ethernet) 카드 등과 같은 내부 및 외부 기술을 포함한다.

[0050] 도시된 네트워크 접속은 단지 예시이며 컴퓨터 사이의 통신 링크를 수립하는 다른 수단이 사용될 수 있음을 인식할 것이다. 당업자는, 컴퓨터(512) 또는 기타 클라이언트 장치는 컴퓨터 네트워크의 일부로서 배치될 수 있음을 인식할 수 있다. 이와 관련하여, 여기 개시된 주제는 여하한 수의 메모리 또는 저장 유닛, 그리고 여하한 수의 저장 유닛 또는 볼륨에 걸쳐 발생하는 여하한 수의 애플리케이션 및 프로세스를 갖는 여하한 컴퓨터 시스템에 관한 것일 수 있다. 여기 개시된 주제의 태양은 원격 또는 로컬 저장소를 갖는, 네트워크 환경에 배치된 서버 컴퓨터 및 클라이언트 컴퓨터를 갖는 환경에 적용될 수 있다. 여기 개시된 주제는 프로그램 언어 기능, 해석 및 실행 기능을 갖는 독립 연산 장치에도 적용될 수 있다.

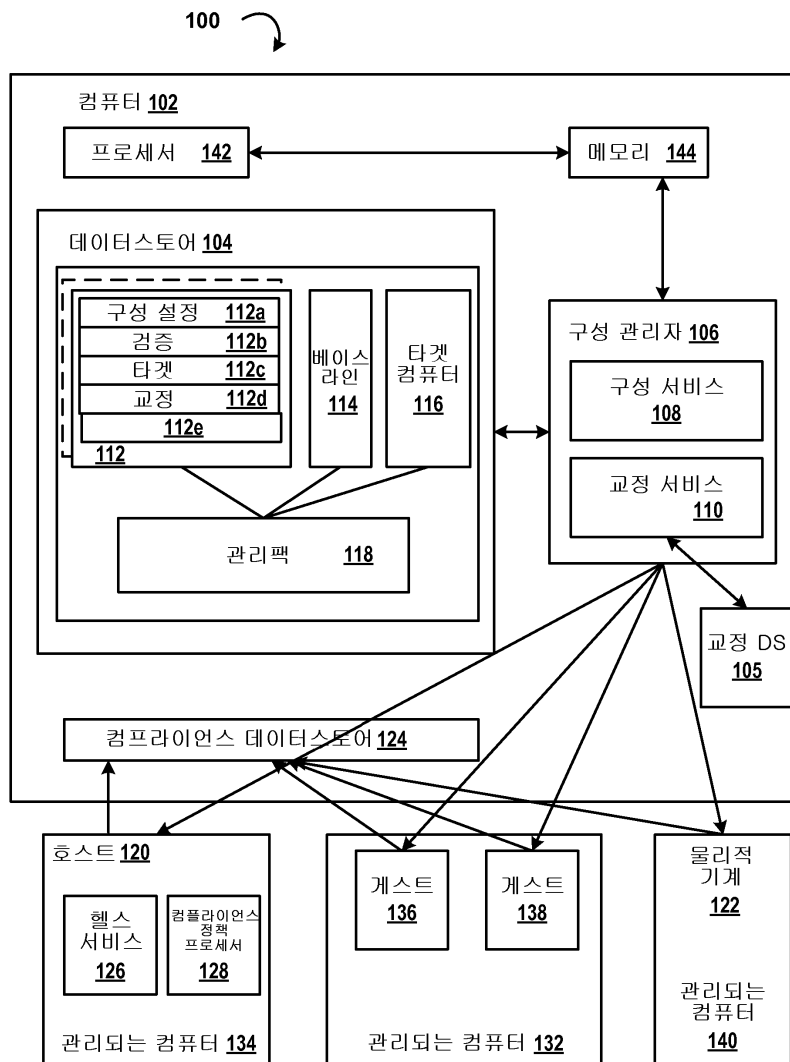
[0051] 여기 설명된 다양한 기술은 하드웨어 또는 소프트웨어와 관련하여, 또는 적당한 경우에는 양자의 결합과 관련하여 구현될 수 있다. 따라서, 여기 개시된 방법 및 장치 또는 그의 특정 태양 또는 부분은 플로피 디스켓, CD-

ROM, 하드 드라이브 또는 여하한 기타 기계-판독가능 저장 매체와 같은 유형의 매체에 포함되는 프로그램 코드(즉, 명령)의 형태를 취할 수 있는데, 여기서, 프로그램 코드가 컴퓨터와 같은 기계로 로딩되고 기계에 의해 실행되는 때에, 기계는 여기 개시된 주제의 태양을 실시하는 장치가 된다. 프로그램가능 컴퓨터 상의 프로그램 코드 실행의 경우에, 연산 장치는 일반적으로 프로세서, 프로세서에 의해 판독가능한 저장 매체(휘발 및 비휘발 메모리 및/또는 저장 요소 포함), 적어도 하나의 입력 장치 및 적어도 하나의 출력 장치를 포함할 것이다. 예를 들어, 데이터 처리 API 등의 사용을 통해, 도메인-특정(domain-specific) 프로그래밍 모델 태양의 생성 및/또는 구현을 사용할 수 있는 하나 이상의 프로그램은, 컴퓨터 시스템과 통신하기 위해 고수준 프로서저(high level procedure) 또는 객체 지향(object-oriented) 프로그램 언어로 구현될 수 있다. 그러나, 프로그램(들)은, 희망에 따라 어셈블리 또는 기계 언어로 구현될 수 있다. 여하한 경우에도, 언어는 컴파일 또는 해석된(interpreted) 언어일 수 있고, 하드웨어 구현과 결합될 수 있다.

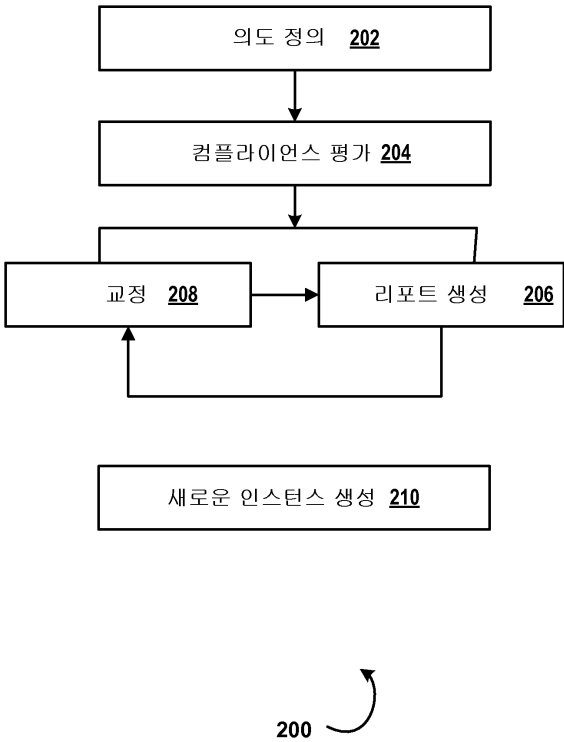
[0052] 여기 개시된 주제가 도면과 관련하여 설명되었지만, 동일한 기능을 다른 방식으로 수행하기 위해 변형이 가해질 수 있음을 이해할 것이다.

도면

도면1



도면2

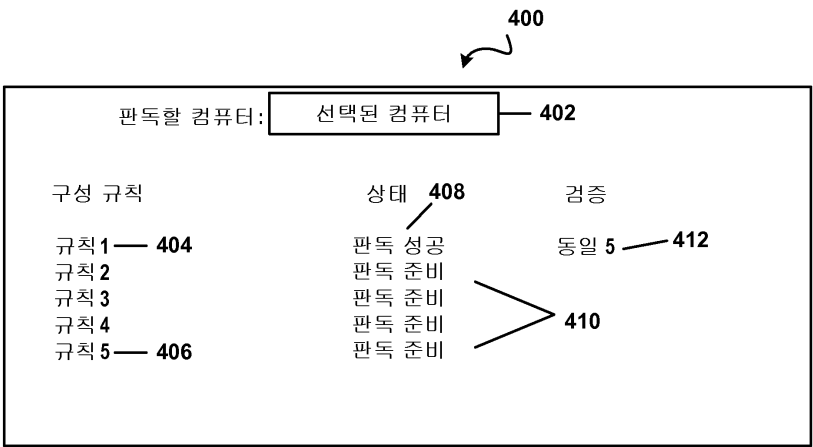


도면3

Figure 3 shows a screenshot of a web application interface. The interface includes a navigation bar with tabs: OVERVIEW, ITEMS, ALERTS, COMPLIANCE, JOBS, PERFORMANCE, and DIAGRAM. The COMPLIANCE tab is selected. The main content area displays a table of compliance items. The table has columns: NAME, STATUS, LAST SCAN, and NEXT SCAN. The table lists several items, including C1_NEWYORKPROD, COMPUTER2_NYP2, PATCH BASELINE, SECURITY BASELINE, SQL BASELINE, COMPUTER3_NYP2, COMPUTER4_NYP2, and COMPUTER5_NYP2. The STATUS column indicates whether each item is COMPLIANT or NON COMPLIANT. The LAST SCAN and NEXT SCAN columns show the dates and times of the last and next scans. The interface also includes a sidebar with a search bar and a list of items. The bottom of the interface shows a pagination bar with the number 300.

NAME	STATUS	LAST SCAN	NEXT SCAN
C1_NEWYORKPROD	COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
COMPUTER2_NYP2	NON COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
PATCH BASELINE	NON COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
SECURITY BASELINE	COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
SQL BASELINE	COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
COMPUTER3_NYP2	COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
COMPUTER4_NYP2	COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM
COMPUTER5_NYP2	NON COMPLIANT	01/07/2010 10:20PM	01/08/2010 8:20PM

도면4



도면5

