SUSE Multi-Linux Manager 5.1

관리 가이드

서문

관리 가이드
SUSE Multi-Linux Manager 5.1

이 가이드에서는 SUSE Multi-Linux Manager 서버의 유지 관리, 모니터링 및 사용자 지정과 같은 관리 작업에 대해 설명합니다.

Publication Date: 2026-04-22

Contents

Chapter 1. 작업

클라이언트 작업은 다음과 같은 다양한 방법으로 관리할 수 있습니다.

- 자동화된 반복 작업을 예약하여 지정된 일정에 따라 Highstate 상태 또는 임의의 사용자 정의 상태 집합을 클라이언트에 적용할 수 있습니다.
- 반복 작업은 개별 클라이언트, 시스템 그룹 내 모든 클라이언트 또는 전체 조직에도 적용 가능합니다.
- 작업 체인을 생성하여 특정 순서로 수행하도록 작업을 설정할 수 있습니다.
  - 작업 체인은 미리 생성 및 편집할 수 있으며 적절한 시간에 실행되도록 일정을 예약할 수 있습니다.
- 하나 이상의 클라이언트에서 원격 명령을 수행할 수도 있습니다.
  - 원격 명령을 사용하면 개별 클라이언트 또는 검색어와 일치하는 모든 클라이언트에 명령을 실행할 수 있습니다.

1.1. 반복 작업

정기적 작업은 개별 클라이언트, 시스템 그룹 또는 조직 내 모든 클라이언트에도 적용 가능합니다.

현재 SUSE Multi-Linux Manager에서 정기적 작업으로 지원하는 작업 유형은 다음과 같습니다.

- **Highstate:** highstate를 실행합니다.
- **사용자 정의 상태:** 일련의 사용자 정의 상태를 실행합니다. 사용자 정의 상태는 SUSE Multi-Linux Manager에서 제공되는 내부 상태이거나 사용자가 생성한 구성 채널일 수 있습니다.

구성 채널에 대한 자세한 내용은 **Client-configuration › Configuration-management**에서 확인할 수 있습니다.

절차: 새 정기적 작업 생성하기

1. 개별 클라이언트에 정기적 작업을 적용하려면 **시스템**으로 이동하여 일정을 구성할 클라이언트를 클릭하고 **정기적 작업** 탭으로 이동합니다.
2. 시스템 그룹에 정기적 작업을 적용하려면 **시스템 › 시스템 그룹**으로 이동하여 일정을 구성할 그룹을 선택한 후 **정기적 작업** 탭으로 이동합니다.
3. **[생성]**을 클릭합니다.
4. **작업 유형** 드롭다운에서 작업 유형을 선택합니다.
5. 새 일정의 이름을 입력합니다.
6. 정기적 작업 빈도를 선택합니다.
   - **매시간:** 각 시간의 분을 입력합니다. 예를 들어, **15**는 매시 15분에 작업을 실행합니다.
   - **매일:** 매일의 시간을 선택합니다. 예를 들어, **01:00**은 SUSE Multi-Linux Manager 서버의 시간대에서 매일 0100시에 작업을 실행합니다.
   - **매주:** 매주 지정된 시간에 작업을 실행할 요일과 시간을 선택합니다.
   - **매월:** 매월 지정된 시간에 작업을 실행할 날짜와 시간을 선택합니다.
   - **사용자 정의 Quartz 형식:** 자세한 옵션을 보려면 사용자 정의 Quartz 스트링을 입력합니다. 예를 들어, 매월

토요일 0215에 정기적 작업을 실행하려면 다음을 입력합니다.

```
0 15 2 ? * 7
```

7. 선택 사항: 테스트 모드에서 일정을 실행하려면 **테스트 모드** 스위치를 토글합니다.
8. **사용자 정의 상태** 유형의 작업일 경우 사용 가능한 상태 목록에서 상태를 선택하고 **[변경사항 저장]**을 클릭합니다. 그러면 일정이 아닌 선택한 현재 상태만 저장됩니다.
9. 다음 패널에서 선택한 상태를 드래그 앤 드롭하여 실행 순서에 넣고 **[확인]**을 클릭합니다.
10. **[일정 생성]**을 클릭하여 저장하고 기존 일정의 전체 목록을 확인합니다.

조직 관리자는 조직의 모든 클라이언트에 대한 정기적 작업을 설정하고 편집할 수 있습니다. **홈 › 내 조직 › 정기적 상태**로 이동하면 조직 전체에 적용되는 모든 반복 작업을 확인 가능합니다.

SUSE Multi-Linux Manager 관리자는 모든 조직의 모든 클라이언트에 대한 정기적 작업을 설정하고 편집할 수 있습니다. **관리 › 조직**으로 이동하여 관리할 조직을 선택한 후 **상태 › 정기적 작업** 탭으로 이동하면 됩니다.

1.2. 동작 체인

클라이언트에서 순차 작업을 여러 개 수행해야 하는 경우 작업 체인을 생성하면 순서대로 작업할 수 있습니다.

기본적으로 대부분의 클라이언트는 명령하는 즉시 작업을 실행합니다. 일부 경우에는 작업에 시간이 오래 걸리므로 이후에 실행된 작업이 실패할 수 있습니다. 예를 들어, 클라이언트에 재부팅을 명령한 후 두 번째 명령을 실행하면 재부팅이 아직 수행 중이므로 두 번째 작업이 실패할 수 있습니다. 작업이 올바른 순서로 발생하도록 하려면 작업 체인을 사용하십시오.

트랜잭션 업데이트 시스템의 경우 재부팅 작업이 남을 때까지 단일 스냅샷 내에서 작업 체인이 실행됩니다. 이로 인해 일부 제한이 발생할 수 있습니다.

자세한 내용은 **Client-configuration › Clients-slemicro**에서 확인할 수 있습니다.

모든 클라이언트에서 작업 체인을 사용할 수 있습니다. 작업 체인에는 다음과 같이 이러한 작업을 원하는 수만큼, 원하는 순서대로 포함할 수 있습니다.

- **시스템 세부 사항 › 원격 명령**
- **시스템 세부 사항 › 시스템 재부팅 예약**
- **시스템 세부 사항 › 상태 › Highstate**
- **시스템 세부 사항 › 소프트웨어 › 패키지 › 목록/제거**
- **시스템 세부 사항 › 소프트웨어 › 패키지 › 설치**
- **시스템 세부 사항 › 소프트웨어 › 패키지 › 업그레이드**
- **시스템 세부 사항 › 소프트웨어 › 패치**
- **시스템 세부 사항 › 소프트웨어 › 소프트웨어 채널**

- **시스템 세부 사항 › 구성**
- **이미지 › 빌드**

작업 체인은 사용자별입니다. Web UI에서 작업 체인을 보려면 작업 체인을 생성한 사용자와 동일한 사용자로 로그인해야 합니다.

절차: 새 작업 체인 생성

1. SUSE Multi-Linux Manager Web UI에서 작업 체인에서 수행할 첫 번째 작업으로 이동합니다. 예를 들어, 클라이언트의 **시스템 세부 정보**로 이동하고 **[시스템 재부팅 예약]**을 클릭합니다.
2. **추가 위치** 필드를 확인하고 추가할 작업 체인 선택:
   - 첫 번째 작업 체인인 경우 **새 작업 체인**을 선택합니다.
   - 작업 체인이 이미 있는 경우 목록에서 선택합니다.
   - 기존 작업 체인이 이미 있지만 새 작업 체인을 생성하려면 새 작업 체인의 이름을 입력하여 생성합니다.
3. 작업을 확인합니다. 작업은 즉시 수행하지는 않으며 새 작업 체인을 생성하고 이를 확인하는 파란색 막대가 화면 상단에 표시됩니다.
4. **추가 위치** 필드를 확인하고 추가할 작업 체인의 이름을 선택하여 작업 체인에 작업 추가를 계속 진행합니다.
5. 작업 추가가 완료되면 **일정 › 작업 체인**으로 이동하여 목록에서 작업 체인을 선택합니다.
6. 작업을 올바른 위치에 끌어다 놓아 작업을 다시 정렬합니다. 클라이언트 작업이 수행될 클라이언트를 살펴보려면 파란색 더하기 기호를 클릭합니다. **[저장]**을 클릭하여 변경 사항을 저장합니다.
7. 작업 체인이 실행될 시간을 예약하고 **[저장 및 예약]**을 클릭합니다. **[저장]** 또는 **[저장 및 예약]**을 클릭하지 않고 페이지를 나가면 저장되지 않은 모든 변경 사항이 삭제됩니다.

작업 체인의 한 작업이 실패하면 작업 체인이 중지되고 더 이상 작업이 실행되지 않습니다.

일정 › 대기 중 작업으로 이동하여 작업 체인에서 예약된 작업을 볼 수 있습니다.

1.3. 원격 명령

이 절차를 사용하여 Salt를 통해 원격 명령을 실행합니다.

클라이언트가 설치된 운영 체제에 적합한 도구 하위 채널에 등록되어 있는지 확인한 후 시작하십시오. 소프트웨어 채널 구독에 대한 자세한 내용은 **Client-configuration › Channels**에서 확인할 수 있습니다.

- 트랜잭션 업데이트 시스템의 경우 단일 스냅샷 내에서 원격 명령이 실행된다는 점에 유의해야 합니다. 이로 인해 일부 제한이 발생할 수 있습니다. 자세한 내용은 **Client-configuration › Clients-slemicro**에서 확인할 수 있습니다.
- 원격 명령은 클라이언트의 **/tmp/** 디렉토리에서 실행됩니다. 원격 명령이 정확하게 작동하도록 하려면 **noexec** 옵션으로 **/tmp**를 마운트하지 마십시오. 자세한 내용은 **Administration › Troubleshooting**에서 확인할 수 있습니다.
- **원격 명령** 페이지에서 실행되는 모든 명령은 클라이언트에서 root를 실행합니다. 와일드카드를 사용하여 여러 시스템에서 명령을 실행할 수 있습니다. 명령을 실행하기 전 항상 주의하십시오.

절차: Salt 클라이언트에서 원격 명령 실행

1. **Salt › 원격 명령**으로 이동합니다.
2. 첫 번째 필드에서 @ 기호 앞에 실행할 명령을 입력합니다.
3. 두 번째 필드에서 @ 기호 뒤에 명령을 실행할 클라이언트를 입력합니다. 개별 클라이언트의 **minion-id**를 입력하거나 와일드카드를 사용하여 클라이언트 범위의 대상을 지정할 수 있습니다.
4. **[대상 찾기]**를 클릭하여 대상으로 지정한 클라이언트를 확인하고 올바른 세부 사항을 사용했는지 확인합니다.
5. **[명령 실행]**을 클릭하여 대상 클라이언트로 명령을 실행합니다.

Chapter 2. Ansible 통합

Ansible은 컴퓨터 클라이언트 시스템을 관리하는 도구입니다. 더 많은 정보를 원하시면, https://www.ansible.com을 참조하십시오.

SUSE Multi-Linux Manager은(는) Ansible 제어 노드 관리를 지원합니다. 자세한 내용은 administration:ansible-setup-control-node.pdf에서 확인할 수 있습니다.

Ansible 제어 노드에서 지원하는 버전은 Ansible 11.3입니다. 운영 체제 공급업체의 공식 리포지토리에서 설치해야 합니다. 예를 들어, SUSE Linux Enterprise 15 SP6 및 SP7에서 Ansible은 **Systems Management Module**을 통해 제공됩니다. SUSE Linux Enterprise 이외의 운영 체제를 실행하는 제어 노드의 경우, 해당 배포판과 함께 제공되는 Ansible을 사용하십시오.

Ansible 소프트웨어는 SUSE Multi-Linux Manager Proxy 및 SUSE Multi-Linux Manager for Retail Branch Server에도 사용할 수 있습니다.

2.1. 기능 개요

SUSE Multi-Linux Manager를 사용하면 시스템 관리자가 Ansible 제어 노드를 작동할 수 있습니다. 지원되는 기능은 다음과 같습니다.

- 인벤토리 파일 검사
- 플레이북 검색
- 플레이북 실행

자세한 내용은 다음을 참조하십시오.

- 인벤토리는 관리형 Ansible 노드의 정렬된 목록입니다. 인벤토리 구성에 대한 자세한 내용은 https://docs.ansible.com/ansible/latest/inventory_guide/intro_inventory.html에서 확인할 수 있습니다.
- 플레이북은 인벤토리 관리 방법을 설명합니다. 플레이북에 대한 자세한 내용은 https://docs.ansible.com/ansible/latest/playbook_guide/playbooks_intro.html에서 확인할 수 있습니다.

2.2. 요구사항 및 기본 구성

Ansible 기능을 사용하려면 기존 Ansible Control Node를 SUSE Multi-Linux Manager 서버에 등록해야 합니다. Web UI에서 등록된 시스템의 **시스템 상세 사항 › 속성** 페이지에서 **추가 기능 시스템 유형** 목록의 **Ansible Control Node** 시스템 유형을 활성화해야 합니다.

Ansible Control Node 시스템 유형을 활성화하면 highstate에 추가하여 **ansible** 패키지를 시스템에 설치하고 **시스템 세부 사항 › Ansible** 탭에서 Ansible 기능을 활성화할 수 있습니다.

다음 단계로 **시스템 세부 사항 › Ansible › 제어 노드** 페이지에서 Ansible 플레이북 디렉토리 및 인벤토리 파일에 대한 경로를 구성합니다. 인벤토리 경로로 표준 Ansible 인벤토리 경로 **/etc/ansible/hosts**를 사용할 수 있습니다. 플레이북 디렉터리로는 플레이북 파일이 저장된 제어 노드의 모든 디렉토리를 사용할 수 있습니다. 플레이북 디렉토리에는 **.yml** 파일 또는 **.yml** 파일이 있는 하위 디렉토리가 포함됩니다.

Ansible 제어 노드 설치 및 설정에 대해서는 **Administration › Ansible-setup-control-node**에서 확인할 수 있습니다.

2.3. 인벤토리 검사

인벤토리 경로를 정의한 후에는 SUSE Multi-Linux Manager를 사용하여 내용을 검사할 수 있습니다.

절차: Web UI에서 인벤토리 검사

1. SUSE Multi-Linux Manager Web UI에서 **시스템 세부 사항 › Ansible › 인벤토리**로 이동합니다.
2. 인벤토리 경로를 클릭하면 제어 노드에서 실시간으로 인벤토리 검사를 실행할 수 있습니다.

2.4. 플레이북 검색

플레이북 디렉토리를 정의한 후 **시스템 세부 사항 › Ansible › 플레이북** 페이지에서 플레이북을 검색할 수 있습니다.

인벤토리 검사와 마찬가지로 플레이북 검색 작업은 제어 노드에서 실시간으로 실행됩니다.

2.5. 플레이북 실행

시스템 세부 정보 › Ansible › 플레이북 페이지에서 플레이북 실행을 예약할 수 있습니다. 실행할 플레이북을 선택한 후 **인벤토리 경로** 드롭다운 메뉴의 **플레이북 실행 예약** 대화 상자에서 실행할 인벤토리 파일을 선택할 수 있습니다. 항목을 선택하지 않으면 제어 노드에 구성된 기본 인벤토리가 사용됩니다. 드롭다운 메뉴는 인벤토리 경로에 정의한 인벤토리와 플레이북 디렉토리에서 로컬로 검색된 인벤토리로 채워집니다. 해당 항목은 플레이북 세부 정보에서 **사용자 정의 인벤토리** 항목으로 표시됩니다. 임의의 인벤토리 경로를 입력할 수도 있습니다.

그런 다음 플레이북 실행 시간을 선택하거나 작업 체인을 선택합니다. 최종적으로 SUSE Multi-Linux Manager는 플레이북을 제어 노드의 작업으로 실행합니다. 작업 결과는 작업 세부 사항 페이지에서 볼 수 있습니다.

2.6. Ansible 제어 노드 설정

Ansible 제어 노드를 설정하려면 SUSE Multi-Linux Manager Web UI에서 다음 단계를 실행합니다.

클라이언트를 Ansible 제어 노드로 구성하려면 해당 시스템에 Ansible 패키지를 설치해야 합니다. 일반적으로 Ansible 패키지는 운영 체제 공급업체의 공식 리포지토리에서 구해야 합니다. 예를 들어, SUSE Linux Enterprise 15 SP6 및 SP7에서는 **Systems Management Module**을 통해 Ansible을 사용할 수 있습니다.

절차: SUSE Linux Enterprise 15 SP6 또는 SP7 시스템에 Ansible 제어 노드 설정

1. SUSE Multi-Linux Manager Web UI에서 **관리자 › 설정 마법사 › 제품**으로 이동하여 **SUSE Linux Enterprise Server 15 SP6 x86_64**(또는 상위 버전)와 **Systems Management Module** 및 필수 **Python 3 Module**이 선택되고 동기화되었는지 확인합니다.
2. SUSE Linux Enterprise 15 SP6(또는 상위 버전) 클라이언트를 배포합니다.

3. SUSE Multi-Linux Manager Web UI에서 클라이언트의 **시스템 › 개요** 페이지로 이동합니다. **소프트웨어 › 소프트웨어 채널**을 선택하고 클라이언트를 **SUSE Linux Enterprise Server 15 SP6 x86_64**(또는 상위 SP 버전) 및 **Systems Management Module**과 **Python 3 Module** 채널을 구독합니다.
4. 클라이언트의 **세부 사항 › 속성**을 선택합니다. **Add-On System Types** 목록에서 **Ansible 제어 노드**를 활성화한 후 [**속성 업데이트**]를 클릭합니다.
5. 클라이언트 개요 페이지로 이동하여 **상태 › Highstate**를 선택하고 [**Highstate 적용**]을 클릭합니다.
6. **이벤트** 탭을 선택하고 highstate의 상태를 확인합니다.

최신 Ansible을 SUSE Linux Enterprise 15 SP4 또는 SP5 클라이언트에 설치하려면 **Python 3 Module**을 활성화해야 합니다.

최신 버전의 Ansible은 더 이상 오래된 Python 버전을 통한 노드 관리를 지원하지 않습니다. 관리형 노드가 여전히 기존 Python 버전으로 기본 설정되어 있는 경우 플레이북 실행 중에 연결 오류가 발생하거나 실패할 수 있습니다. 이 문제를 해결하려면 가능한 경우 관리형 노드에서 Python을 업그레이드하고 Ansible 인벤토리 또는 구성에서 올바른 Python 인터프리터를 설정해야 합니다.

2.6.1. Ansible 인벤토리 파일 생성

Ansible Integration 도구는 플레이북을 인벤토리 파일로 배포합니다. _테이블 1_에 나열된 각 운영 체제에 대해 인벤토리 파일을 하나씩 생성합니다.

절차: Ansible 인벤토리 파일 생성

1. Ansible에서 관리할 인벤토리 파일에 호스트를 생성 및 추가합니다. Ansible 인벤토리의 기본 경로는 **/etc/ansible/hosts**입니다.

목록 1. 인벤토리 예시

```
client240.mgr.example.org
client241.mgr.example.org
client242.mgr.example.org
client243.mgr.example.org
ansible_ssh_private_key_file=/etc/ansible/some_ssh_key

[mygroup1]
client241.mgr.example.org
client242.mgr.example.org

[mygroup2]
client243.mgr.example.org

[all:vars]
```

```
ansible_ssh_public_key_file=/etc/ansible/my_ansible_private_key
```

2. SUSE Multi-Linux Manager Web UI의 **Ansible** 탭에서 **Ansible › 제어 노드**로 이동하여 제어 노드에 인벤토리 파일을 추가합니다.
3. **Playbook Directories** 섹션에서 **Add a Playbook Directories** 필드에 **/usr/share/scap-security-guide/ansible**을 추가한 후 [**저장**]을 클릭합니다.
4. **인벤토리 파일**에서 **인벤토리 파일 추가** 필드에 인벤토리 파일 위치를 추가하고 [**저장**]을 클릭합니다.

목록 2. 예제

```
/etc/ansible/sles15
/etc/ansible/sles12
/etc/ansible/centos7
```

더 많은 플레이북 예시는 https://github.com/ansible/ansible-examples에서 확인할 수 있습니다.

절차: Ansible 노드와 통신 설정하기

1. 인벤토리에서 사용 중인 SSH 키를 생성합니다.

```
ssh-keygen -f /etc/ansible/my_ansible_private_key
```

2. 생성된 SSH 키를 Ansible 관리 클라이언트에 복사합니다. 예시는 다음과 같습니다.

```
ssh-copy-id -i /etc/ansible/my_ansible_public_key root@client240.mgr.example.org
```

3. **/etc/ansible/ansible.cfg**에 다음과 같이 개인 키를 선언합니다.

```
private_key_file = /etc/ansible/my_ansible_private_key
```

my_ansible_private_key를 개인 키가 포함된 파일의 이름으로 바꿉니다.

4. 제어 노드에서 다음 명령을 실행하여 Ansible 작동 여부를 테스트하십시오.

```
ansible all -m ping
ansible mygroup1 -m ping
ansible client240.mgr.example.org -m ping
```

이제 수정을 실행할 수 있습니다. 자세한 내용은 **Administration › Ansible-compliance-as-code**에서 확인할 수 있습니다.

2.7. 코드형 규정 준수

이 문서에서는 Ansible Playbook을 사용한 코드형 규정 준수 수정을 실행하는 방법에 대한 통찰력을 제공합니다.

bash 스크립트를 사용해 코드형 규정 준수 수정을 실행하는 방법에 대한 자세한 내용은 수정을 참조하십시오.

2.7.1. SCAP 보안 가이드 패키지 설치

수정을 실행하려면 Ansible 제어 노드에 SCAP 보안 가이드 패키지를 설치해야 합니다.

절차: SCAP 보안 가이드 패키지 설치

1. **시스템 › 개요**에서 클라이언트를 선택한 후, **소프트웨어 › 패키지 › 설치**를 클릭합니다.
2. **scap-security-guide**를 검색하여 시스템에 적합한 패키지를 설치합니다. 패키지 배포 요구 사항은 다음 테이블을 참조하십시오.

표 1. SCAP 보안 가이드 패키지 요구 사항

패키지 이름	지원되는 시스템
scap-security-guide	openSUSE, SLES12, SLES15
scap-security-guide-redhat	CentOS 7, CentOS 8, Fedora, Oracle Linux 7, Oracle Linux 8, RHEL7, RHEL8, RHEL9, Red Hat OpenStack Platform 10, Red Hat OpenStack Platform 13, Red Hat Virtualization 4, Scientific Linux
scap-security-guide-debian	Debian 12
scap-security-guide-ubuntu	Ubuntu 20.04, Ubuntu 22.04

2.7.2. Ansible 플레이북을 사용한 수정

Ansible 제어 노드가 필요합니다. 자세한 내용은 **Administration › Ansible-setup-control-node**에서 확인할 수 있습니다.

다음 절차는 Ansible 플레이북을 사용하여 교정을 실행하는 과정을 안내합니다.

절차: Ansible 플레이북을 사용한 수정 실행

1. 제어 노드 시스템 메뉴에서 **Ansible › 플레이북**을 선택하고 플레이북을 클릭합니다. 예는 다음과 같습니다.

```
sle15-playbook-stig.yml
```

2. 플레이북을 실행하려면 클라이언트의 **인벤토리 경로**를 선택합니다. 예는 다음과 같습니다.

```
/etc/ansible/sles15
```

[일정]을 클릭합니다.

3. **이벤트** 탭에서 예약된 이벤트의 상태를 확인합니다.

플레이북이 다른 디렉토리에 있는 경우, **Ansible 제어 노드 설정** 링크를 따라 추가하는 방법을 확인할 수 있습니다.

Chapter 3. 인증 방법

SUSE Multi-Linux Manager는 다양한 인증 방법을 지원합니다. 이 섹션에서는 PAM(pluggable authentication modules) 및 SSO(싱글 사인온)에 대해 설명합니다.

3.1. SSO(싱글 사인온)로 인증

SUSE Multi-Linux Manager는 SAML(Security Assertion Markup Language) 2 프로토콜을 구현하여 SSO(싱글 사인온)를 지원합니다.

싱글 사인온은 사용자가 한 개의 자격 증명으로 여러 애플리케이션에 액세스할 수 있도록 해주는 인증 프로세스입니다. SAML은 인증 및 권한 부여 데이터를 교환하기 위한 XML 기반 표준입니다. SAML ID 서비스 공급자(IdP)는 SUSE Multi-Linux Manager 등 서비스 공급자(SP)에 인증 및 권한 부여 서비스를 제공합니다. SUSE Multi-Linux Manager는 싱글 사인온을 위해 활성화해야 하는 엔드포인트 세 개를 노출합니다.

SUSE Multi-Linux Manager의 SSO가 지원하는 사항:

- SSO로 로그인합니다.
- 서비스 제공자 시작 싱글 로그아웃(SLO) 및 ID 서비스 제공자 싱글 로그아웃 서비스(SLS)로 로그아웃합니다.
- 어설션 및 nameID 암호화.
- 어설션 서명.
- AuthNRequest, LogoutRequest 및 LogoutResponses 포함 메시지 서명.
- 어설션 소비자 서비스 엔드포인트를 활성화합니다.
- 싱글 로그아웃 서비스 엔드포인트를 활성화합니다.
- SP 메타데이터(서명 가능한)를 게시합니다.

SUSE Multi-Linux Manager의 SSO는 다음을 지원하지 않습니다.

- ID 서비스 제공자(IdP)를 위한 제품 선택 및 구현.
- 기타 제품에 대한 SAML 지원(해당 제품 설명서 확인).

SSO 구현의 예는 **Administration › Auth-methods-sso-example**에서 확인할 수 있습니다.

기본 인증 방법에서 싱글 사인온으로 변경하면 새 SSO 자격 증명이 Web UI에만 적용됩니다. **mgr-sync** 또는 **spacecmd**와 같은 클라이언트 도구는 기본 인증 방법으로만 계속해서 작동합니다.

3.1.1. 선행 조건

시작하기 전 이러한 파라미터를 사용하여 외부 ID 서비스 제공자를 구성해야 합니다. 지침은 IdP 설명서를 확인하십시오.

IdP 사용자와 SUSE Multi-Linux Manager 사용자 간의 매핑은 SAML:Attribute에 지정됩니다. SAML:Attribute은 IdP에서 구성해야 하며 SAML 인증에서 SUSE Multi-Linux Manager에 전달되어야 합니다. 특성의 이름은 **uid**여야 하며 로그인 후 매핑된 SUSE Multi-Linux Manager 사용자를 포함해야 합니다. SUSE Multi-Linux Manager을(를) 만든 후 통합 인증을 활성화해야

합니다.

필요한 엔드포인트는 다음과 같습니다.

- 어설션 소비자 서비스(또는 ACS): 서비스 제공자에 대한 세션을 연결하기 위해 SAML 메시지를 수락하는 엔드포인트입니다. SUSE Multi-Linux Manager의 ACS 엔드포인트: https://server.example.com/rhn/manager/sso/acs
- 싱글 로그아웃 서비스(또는 SLS): IdP에서 로그아웃 요청을 시작하는 엔드포인트입니다. SUSE Multi-Linux Manager의 SLS 엔드포인트: https://server.example.com/rhn/manager/sso/sls
- 메타데이터: SAML에 대한 SUSE Multi-Linux Manager 메타데이터를 검색하기 위한 엔드포인트입니다. SUSE Multi-Linux Manager의 메타데이터 엔드포인트: https://server.example.com/rhn/manager/sso/metadata

사용자 **orgadmin**을 사용한 IdP 인증에 성공하면 **orgadmin** 사용자가 SUSE Multi-Linux Manager에 존재하는 경우 SUSE Multi-Linux Manager에 **orgadmin** 사용자로 로그인됩니다.

3.1.2. SSO 사용

SSO 사용은 다른 유형의 인증과 상호 배타적이며 활성화 또는 비활성화됩니다. SSO는 기본적으로 비활성화되어 있습니다.

서버 컨테이너 내에서 단계를 실행하기 전에 **mgrctl term**을 사용합니다.

절차: SSO 활성화하기

1. 아직 사용자가 SUSE Multi-Linux Manager에 없는 경우 먼저 생성해야 합니다.
2. **/etc/rhn/rhn.conf**를 편집하고 파일 끝에 다음 라인을 추가합니다.

```
java.sso = true
```

3. **/usr/share/rhn/config-defaults/rhn_java_sso.conf**에서 사용자 정의할 파라미터를 찾습니다. 사용자 정의할 파라미터를 **/etc/rhn/rhn.conf**에 삽입하고 **java.sso**를 접두사로 붙입니다. 예를 들어, **/usr/share/rhn/config-defaults/rhn_java_sso.conf**에서 다음을 검색합니다.

```
onelogin.saml2.sp.assertion_consumer_service.url = https://YOUR-PRODUCT-HOSTNAME-OR-IP/rhn/manager/sso/acs
```

사용자 정의하려면 **/etc/rhn/rhn.conf**에서 옵션 이름에 **java.sso.**를 접두사로 붙여 해당 옵션을 생성합니다.

```
java.sso.onelogin.saml2.sp.assertion_consumer_service.url = https://YOUR-PRODUCT-HOSTNAME-OR-IP/rhn/manager/sso/acs
```

변경해야 하는 모든 항목을 찾으려면 파일에서 **YOUR-PRODUCT** 및 **YOUR-IDP-ENTITY** 자리 표시자를 검색합니다. 모든 파라미터에는 해당 의미에 대한 간략한 설명이 함께 제공됩니다.

4. Spacewalk 서비스를 다시 시작하여 변경 사항을 적용합니다.

```
mgradm restart
```

SUSE Multi-Linux Manager URL로 이동하면 인증이 요청된 SSO용 IdP로 리디렉션됩니다. 인증에 성공하면 인증된 사용자로 로그인한 SUSE Multi-Linux Manager Web UI로 리디렉션됩니다. SSO를 사용한 로그인에서 문제가 발생하면 SUSE Multi-Linux Manager 로그에서 자세한 내용을 확인할 수 있습니다.

3.1.3. 예시 SSO 구현

이 예에서 SSO는 SUSE Multi-Linux Manager로 3개의 엔드포인트를 노출하고 Keycloak 21.0.1 이상을 ID 서비스 공급자(IdP)로 사용하여 구현됩니다.

Keycloak IdP를 설치한 후 SUSE Multi-Linux Manager 서버를 설정하여 시작하십시오. 그런 다음 엔드포인트를 Keycloak 클라이언트로 추가하고 사용자를 생성할 수 있습니다.

이 예는 설명 목적으로만 제공됩니다. SUSE는 타사 ID 서비스 제공자를 권장하거나 지원하지 않으며 Keycloak과 제휴하지 않습니다. Keycloak 지원에 대해서는 https://www.keycloak.org/를 참조하십시오.

Keycloak을 시스템에 직접 설치하거나 컨테이너에서 실행할 수 있습니다. 이 예에서는 Podman 컨테이너에서 Keycloak을 실행합니다. Keycloak 설치에 대한 자세한 내용은 https://www.keycloak.org/guides#getting-started에서 Keycloak 설명서를 참조하십시오.

절차: ID 서비스 공급자 설정

1. Keycloak 설명서에 따라 Podman 컨테이너에 Keycloak을 설치합니다.
2. 프로세스를 계속 실행하려면 **-td** 인수를 사용하여 컨테이너를 실행하십시오.

```
podman run -td --name keycloak -p 8080:8080 -e KEYCLOAK_USER=admin -e
KEYCLOAK_PASSWORD=admin quay.io/keycloak/keycloak:21.0.1
```

3. **관리** 사용자로 Keycloak Web UI에 로그인하고 다음 세부 정보를 사용하여 인증 영역을 생성합니다.
   - **이름** 필드에 영역 이름을 입력합니다. 예를 들어, **MLM**입니다.
   - **엔드포인트** 필드에서 **SAML 2.0 ID 공급자 메타 데이터** 링크를 클릭합니다. 그러면 SUSE Multi-Linux Manager 구성 파일에 복사할 엔드포인트와 인증서가 표시되는 페이지로 이동합니다.

Keycloak을 설치하고 영역을 생성한 후에는 SUSE Multi-Linux Manager 서버를 준비할 수 있습니다.

절차: SUSE Multi-Linux Manager 서버 설정

1. SUSE Multi-Linux Manager 서버에서 **/etc/rhn/rhn.conf** 구성 파일을 열고 이러한 파라미터를 편집합니다. **<FQDN_MLM>**을 SUSE Multi-Linux Manager 설치의 정규화된 도메인 이름으로 바꿉니다.

```
java.sso.onelogin.saml2.sp.entityid                       =
https://<FQDN_MLM>/rhn/manager/sso/metadata
java.sso.onelogin.saml2.sp.assertion_consumer_service.url =
https://<FQDN_MLM>/rhn/manager/sso/acs
java.sso.onelogin.saml2.sp.single_logout_service.url      =
https://<FQDN_MLM>/rhn/manager/sso/sls
```

2. 구성 파일에서 **<FQDN_IDP>**를 Keycloak 서버의 정규화된 도메인 이름으로 바꿉니다. **<REALM>**을 인증 영역(예: **MLM**)으로 바꿉니다.

```
java.sso.onelogin.saml2.idp.entityid                   =
http://<FQDN_IDP>:8080/realms/<REALM>
java.sso.onelogin.saml2.idp.single_sign_on_service.url =
http://<FQDN_IDP>:8080/realms/<REALM>/protocol/saml
java.sso.onelogin.saml2.idp.single_logout_service.url  =
http://<FQDN_IDP>:8080/realms/<REALM>/protocol/saml
```

3. IdP 메타데이터에서 공용 x509 인증서를 찾습니다. http://<FQDN_IDP>:8080/realms/<REALM>/protocol/saml/descriptor의 형식을 사용하십시오. 이 구성 파일에서 IdP의 공용 x509 인증서를 다음과 같이 지정합니다.

```
java.sso.onelogin.saml2.idp.x509cert = -----BEGIN CERTIFICATE----- <CERTIFICATE>
-----END CERTIFICATE-----
```

SSO를 활성화한 후 SUSE Multi-Linux Manager에서 **rhn.conf**의 예는 다음과 같습니다.

```
java.sso = true


# SAMLv2 프로토콜을 통한 SSO(Single Sign-On) 구성 파일입니다.
# SSO를 활성화하려면 /etc/rhn/rhn.conf에서 java.sso = true로 설정하십시오.
#
# 필수 변경 사항: 이 파일에서 다음을 검색합니다.
# - YOUR-PRODUCT
# - YOUR-IDP-ENTITY
#
# 모든 파라미터에 대한 자세한 내용은
# 제품 설명서 및 이 파일의 인라인 주석을 참조하십시오.
#
#
#
#
# 'strict'가 True인 경우 Java Toolkit는 서명되거나 암호화되어야 하지만
# 서명되지 않거나 암호화되지 않은 메시지를 거부합니다.
# 또한, SAML을 엄격히 따르지 않는 메시지도 거부합니다.
#
# 경고: 프로덕션에서 이 파라미터 설정 파라미터는 "true"로 설정되어야 합니다.
# 그렇지 않으면 환경이 안전하지 않고 공격에 노출됩니다.
# 디버그 모드 활성화(오류 출력)
# SP 항목의 식별자(URI여야 함)
java.sso.onelogin.saml2.sp.entityid =
https://MLMserver.example.org/rhn/manager/sso/metadata

# <AuthnResponse> 메시지가 요청자(이 경우 SP)에게 반환되어야 하는
# 위치와 방법에 대한 정보를 지정합니다.
# IdP의 <Response>가 반환될 URL 위치
java.sso.onelogin.saml2.sp.assertion_consumer_service.url =
https://MLMserver.example.org/rhn/manager/sso/acs

# <Logout Response> 메시지가 요청자(이 경우 SP)에게 반환되어야 하는
# 위치와 방법에 대한 정보를 지정합니다.
java.sso.onelogin.saml2.sp.single_logout_service.url =
https://MLMserver.example.org/rhn/manager/sso/sls

# IdP 항목의 식별자(URI여야 함)
```

```
java.sso.onelogin.saml2.idp.entityid = http://idp.example.org:8080/realms/MLM

# IdP의 SSO 엔드포인트 정보입니다. (인증 요청 프로토콜)
# SP가 인증 요청 메시지를 보낼 IdP의 URL 대상
java.sso.onelogin.saml2.idp.single_sign_on_service.url =
http://idp.example.org:8080/realms/MLM/protocol/saml

# IdP의 SLO 엔드포인트 정보입니다.
# SP가 SLO 요청을 보낼 IdP의 URL 위치
java.sso.onelogin.saml2.idp.single_logout_service.url =
http://idp.example.org:8080/realms/MLM/protocol/saml

# IdP의 공개 x509 인증서
java.sso.onelogin.saml2.idp.x509cert = -----BEGIN CERTIFICATE-----
MIIClzCCAX8CBgGC+tPbVjANBgkqhkiG9w0BAQsFADAPMQ0wCwYDVQQDDARTVU1BMB4XDTIyMDkwMTIwNTEwNFoXDT
MyMDkwMTIwNTI0NFowDzENMAsG
A1UEAwwEU1VNQTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMNSWJAalB5mShTkMBO5mrsOosyheEL8/A
37WvuqDPwwEfm4x0cG7gmMHvONxYXZk+LRyzoQl2sBrNFrbMuwu5dnah5ZSMxQyUu697S280m4vIiegGaFdbgH+g4F
GBu
eSis1ssMzTcES+NUuI7pLkMLNmSQtncESnoL9q2SyeQSwYtr5dz1ydl6IzjwtaWeyQ9EGJNtJtLk3U4+arLPCpHAwq
FAnLO9NeYcRDNUKhNBs1v5mHP+L066PZu1/DkE0mSgy/+qXaS0CgZVKqz8qB+bvHVuAq9W60g1CjqZKbwvPu72p/7+
d8z
9DxXPIZ1uxdqn19q/kLEP2TYLtgQobSHECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAga+raLMJDo/P/yN1Z6SGGocK
227WFqovBiE/mLYlp5Ff0+0jS1US1plSppJ94xOr8j0m7HW0Wu5xCz6oOhzXTEtnfIbeRyr1Rms3BWdxyXgQ9bWUeZ
MWZ
HfDkTbhgRRmjDEwSSfEXRKQNvw41CpnlB36I0++ejgGnjDvH7BbkCaoW55JF5j6DT/WYR0n7MkEl2Ova9CH0e9X7Gn
y8iOAg26oziy06uy3P/lx9Z9RmHnvpvN/Q34SGEq9z/HlQVuP12UPj//iT21Jc17OOZFsZQXlGFTG6bXKmO42W8FdU
DJU
ONoXZgjMb3eC7U691YyeowoqTY7mJKxNPprYY/lL0w== -----END CERTIFICATE-----


# 조직
java.sso.onelogin.saml2.organization.name = SUSE Manager admin
java.sso.onelogin.saml2.organization.displayname = SUSE Manager admin
java.sso.onelogin.saml2.organization.url = https://MLMserver.example.org
java.sso.onelogin.saml2.organization.lang =

# Contacts
java.sso.onelogin.saml2.contacts.technical.given_name = SUSE Manager admin
java.sso.onelogin.saml2.contacts.technical.email_address = MLM@example.org
java.sso.onelogin.saml2.contacts.support.given_name = SUSE Manager admin
java.sso.onelogin.saml2.contacts.support.email_address = MLM@example.org
```

SUSE Multi-Linux Manager 엔드포인트를 Keycloak에 추가할 수 있습니다. Keycloak은 엔드포인트를 클라이언트로 참조합니다.

절차: 엔드포인트를 클라이언트로 추가

1. Keycloak Web UI에서 다음 세부 사항을 사용하여 새 클라이언트를 생성합니다.
   - **클라이언트 유형** 필드에서 **SAML**을 선택합니다.
   - **클라이언트 ID** 필드에서 서버 구성 파일에 지정된 엔드포인트를 **java.sso.onelogin.saml2.idp.entityid**로 입력합니다. 예: https://<FQDN_MLM>/rhn/manager/sso/metadata.
2. **설정** 탭에서 다음 세부 정보를 사용하여 클라이언트를 미세 조정합니다.
   - **서명 어설션** 스위치를 **켜짐**으로 토글합니다.
   - **서명 알고리즘** 필드에서 **RSA_SHA1**을 선택합니다.

- **SAML 서명 키 이름** 필드에서 **키 ID**를 선택합니다.

3. **키** 탭에서:
   - **클라이언트 서명 필요**를 **끄기**로 설정합니다.
4. **세분화된 SAML 엔드포인트 구성** 섹션의 **고급** 탭에서 다음 세부 정보를 사용하여 엔드포인트 두 개를 추가합니다.
   - 두 **어설션 소비자 서비스** 필드에, 서버 구성 파일에 지정된 엔드포인트를 **java.sso.onelogin.saml2.sp.assertion_consumer_service.url**로 입력합니다. 예: https://<FQDN_MLM>/rhn/manager/sso/acs.
   - 두 **로그아웃 서비스** 필드에, 서버 구성 파일에 지정된 엔드포인트를 **java.sso.onelogin.saml2.sp.single_logout_service.url**로 입력합니다. 예: https://<FQDN_MLM>/rhn/manager/sso/sls.

엔드포인트를 클라이언트로 추가하면 클라이언트 범위를 구성하고 Keycloak과 SUSE Multi-Linux Manager 간에 사용자를 매핑할 수 있습니다.

절차: 클라이언트 범위 및 매퍼 구성

1. Keycloak Web UI에서 **클라이언트 › 클라이언트 범위** 탭으로 이동하여 **role_list**를 기본 클라이언트 범위로 할당합니다.
2. **클라이언트_범위 › 매퍼** 탭으로 이동하여 기본값을 사용해 사용자 속성 **uid**에 대한 매퍼를 추가합니다. 이 SAML 특성은 SUSE Multi-Linux Manager에 필요합니다.
3. **클라이언트_범위 › 매퍼** 로 이동하여 **역할_목록** 매퍼를 클릭합니다. **단일 역할 속성**을 **켜기**로 설정합니다.
4. **사용자 › 관리자** 섹션으로 이동하여 관리 사용자를 생성합니다. 이 사용자는 SUSE Multi-Linux Manager 관리 사용자와 일치하지 않아도 됩니다.
5. **사용자 › 역할 매핑** 탭으로 이동하여 SUSE Multi-Linux Manager 관리 사용자의 사용자 이름과 일치하는 값을 갖는 **uid**라는 속성을 추가합니다.
6. **사용자 › 자격 증명** 탭으로 이동하여 SUSE Multi-Linux Manager 관리 사용자가 사용하는 것과 동일한 비밀번호를 설정합니다. . Save your changes.

구성을 완료하면 설치가 예상대로 작동하는지 테스트할 수 있습니다. SUSE Multi-Linux Manager 서버를 재시작하여 변경 사항을 적용하고 SUSE Multi-Linux Manager Web UI로 이동합니다. 설치가 올바르게 작동하면 성공적으로 인증할 수 있는 Keycloak SSO 페이지로 리디렉트됩니다.

3.2. PAM으로 인증

SUSE Multi-Linux Manager은(는) SSSD를 사용하는 플러그형 인증 모듈(PAM)을 사용하여 네트워크 기반 인증 시스템을 지원합니다. PAM은 SUSE Multi-Linux Manager을(를) 중앙집중식 인증 메커니즘과 통합할 수 있는 라이브러리 모음으로, 여러 개의 비밀번호를 기억할 필요가 없습니다. SUSE Multi-Linux Manager은(는) LDAP, Kerberos 및 기타 네트워크 기반 인증을 지원합니다.

3.2.1. SSSD 구성

절차: SSSD 구성

1. SUSE Multi-Linux Manager Web UI에서 **사용자 › 사용자 생성**으로 이동하고 새로운 또는 기존 사용자를 활성화하여 PAM으로 인증합니다.

   사용자 이름에는 영숫자 외에 **-**, **_**, **.**, **@**을 사용할 수 있습니다.

2. **PAM(Pluggable Authentication Modules)** 확인란을 선택하십시오.
3. 서버 컨테이너에서 SSSD를 구성하십시오. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 루트 권한으로 서버 컨테이너를 입력하십시오.

```
mgrctl term
```

4. 컨테이너 내에서 다음 단계를 실행하십시오.
   a. 구성에 따라 **/etc/sssd/sssd.conf**를 편집합니다. 예제는 administration:auth-methods-pam.pdf에서 확인할 수 있습니다.
   b. 완료하면 컨테이너를 종료:

```
exit
```

5. 다음을 사용하여 SUSE Multi-Linux Manager을(를) 다시 시작합니다.

```
mgradm restart
```

SUSE Multi-Linux Manager Web UI에서 비밀번호를 변경하면 SUSE Multi-Linux Manager 서버의 로컬 비밀번호만 변경됩니다. 해당 사용자에 대해 PAM이 활성화된 경우 로컬 비밀번호가 사용되지 않을 수 있습니다. 예를 들어, 위의 예에서 Kerberos 비밀번호는 변경되지 않습니다. 이러한 사용자의 경우 네트워크 서비스의 비밀번호 변경 메커니즘을 사용하여 비밀번호를 변경하십시오.

PAM 구성에 대한 자세한 내용은 SUSE Linux Enterprise Server 보안 가이드를 참조하십시오. 이 보안 가이드에는 다른 네트워크 기반 인증 방법에도 적용되는 일반적인 예제도 포함되어 있습니다. 또한 AD(Active Directory) 서비스를 구성하는 방법도 설명합니다. 자세한 내용은 https://documentation.suse.com/sles/15-SP7/html/SLES-all/part-auth.html을 참조하십시오.

3.2.1.1. Active Directory와 LDAP 통합 예제

Active Directory와 LDAP를 직접 통합하려면 다음 예제를 사용하면 됩니다.

코드 조각에서 사용자 환경에 따라 다음 자리 표시자를 변경합니다.

$domain

도메인 이름

$ad_server

$domain $uyuni-hostname에서 자동 감지되지 않는 경우 AD 서버의 FQDN입니다. 이는 이 AD 클라이언트가 알아야 하는 컴퓨터의 이름입니다. 설정하지 않으면 **uyuni-server.mgr.internal**로 설정됩니다.

/etc/sssd/sssd.conf에 대한 예시 코드 조각:

```
[sssd]
config_file_version = 2
services = nss, pam
domains = $domain

[nss]

[pam]

[domain/$domain]
id_provider = ad
chpass_provider = ad
access_provider = ad
auth_provider = ad

ad_domain = $domain
ad_server = $ad_server
ad_hostname = $uyuni-hostname

ad_gpo_map_network = +susemanager

krb5_keytab = FILE:/etc/rhn/krb5.conf.d/krb5.keytab
krb5_ccname_template = FILE:/tmp/krb5cc_%{uid}
```

Chapter 4. 백업 및 복구

이 장에는 백업해야 할 파일에 대한 정보가 포함되어 있습니다. 기본 제공 백업 및 복원 솔루션(**mgradm backup**)을 사용하여 SUSE Multi-Linux Manager의 백업을 생성하십시오. 이 장은 시스템 장애 발생 시 백업에서 복원하는 방법에 대한 정보로 마무리됩니다.

SUSE Multi-Linux Manager에서는 설치된 프로그램 및 구성뿐만 아니라 데이터베이스를 활용하므로 설치의 모든 구성 요소를 백업하는 것이 중요합니다. 데이터 손실을 방지하고 빠르게 복구할 수 있도록 SUSE Multi-Linux Manager 설치를 정기적으로 백업하십시오.

백업 방법과 관계없이 현재 설치 작업 중 사용하는 공간의 세 배 이상을 사용할 수 있어야 합니다. 공간이 부족하면 백업이 실패할 수 있으므로 수시로 확인하십시오.

4.1. smdba를 사용하여 기존 방법 비활성화

SUSE Multi-Linux Manager 5.1 제품을 처음부터 설치했다면 이 섹션을 건너뛰십시오.

기본 제공 솔루션이 도입되어 **smdba** 백업 도구를 사용하는 기존 방식은 더 이상 사용되지 않습니다. **smdba**를 사용하던 기존 시스템에서 새 솔루션으로 마이그레이션한 경우, 기존 기능을 비활성화하고 기존 백업 아카이브를 제거해야 합니다.

smdba를 비활성한 후에 마이그레이션(권장)하거나, 마이그레이션된 SUSE Multi-Linux Manager 5.1 시스템에서 나중에 비활성화하십시오.

절차: 마이그레이션 이전에 설치된 smdba를 사용한 기존 방법 비활성화

이 절차는 smdba가 아직 설치된 상태에서만 작동합니다.

1. SUSE Manager 4.3(비컨테이너화 설치) 또는 SUSE Manager 5.0(컨테이너화 설치)에서는 명령이 다릅니다(따라서 5.0 또는 마이그레이션 이전에 4.3 사용).

   SUSE Manager 4.3

   명령줄에서 루트 권한으로 다음을 실행합니다.

```
smdba backup-hot --enable=off
```

   SUSE Manager 5.0

   컨테이너 호스트의 명령줄에서 루트 권한으로 다음을 실행합니다.

```
mgrctl exec -- smdba backup-hot --enable=off
```

   이 작업을 수행하면 **/var/lib/pgsql/data/postgresql.conf** 내의 **archive_command**가 다음과 같이 변경됩니다.

```
archive_command = '/bin/true'
```

이제 기존 시스템을 SUSE Multi-Linux Manager 5.1(으)로 마이그레이션할 준비가 되었습니다.

절차: 마이그레이션 후 SUSE Multi-Linux Manager 5.1에서 기존 방법 비활성화

이 절차는 마이그레이션 후 smdba를 더 이상 사용할 수 없는 경우에 사용하십시오.

1. 컨테이너 호스트에서 루트 권한으로 **/var/lib/containers/storage/volumes/var-pgsql/_data/postgresql.conf** 파일을 편집하고 다음 옵션을 설정합니다.

```
archive_mode = off
archive_command = '/bin/true'
```

2. 컨테이너를 재시작합니다.

```
mgradm restart
```

4.2. SUSE Multi-Linux Manager 백업

SUSE Multi-Linux Manager 설치를 가장 포괄적으로 백업하는 방법은 **mgradm backup create** 명령을 사용하는 것입니다. 이 방법을 사용하면 백업 관리 시간을 절약할 수 있으며 오류가 발생한 경우 더 빠르게 다시 설치 및 재동기화할 수 있습니다. 그러나 이 방법은 상당히 큰 규모의 디스크 공간이 필요하고 백업을 수행하는 데 시간이 오래 걸릴 수 있습니다.

mgradm backup create 명령은 디렉토리에 백업을 수행합니다. 이 디렉토리는 로컬 저장소 또는 마운트된 원격 저장소일 수 있습니다.

mgradm backup create 명령을 사용하면 백업 내용을 다양하게 사용자 정의할 수 있습니다. 사용 가능한 모든 옵션은 **mgradm backup create --help**를 참조하십시오.

4.2.1. SUSE Multi-Linux Manager 전체 백업

SUSE Multi-Linux Manager의 전체 백업은 다음 구성 요소를 백업하는 것으로 구성됩니다.

- SUSE Multi-Linux Manager 볼륨
- 데이터베이스 볼륨
- podman 네트워크 구성
- podman 비밀
- SUSE Multi-Linux Manager systemd 서비스
- SUSE Multi-Linux Manager 컨테이너 이미지

SUSE Multi-Linux Manager 서비스는 전체 백업을 생성하는 데 소요되는 시간 동안 자동으로 중지됩니다. 다운타임이 상당할 수 있습니다. 백업이 완료되면 서비스가 자동으로 다시 시작됩니다.

절차: mgradm backup create로 전체 백업 만들기

1. 컨테이너 호스트에서 루트 권한으로 다음을 통해 백업:

```
mgradm backup create $path
```

$path를 백업 위치의 경로로 바꿉니다.

4.2.2. SUSE Multi-Linux Manager 부분 백업

mgradm backup create 도구를 사용하면 부분 백업을 생성할 수 있습니다. 개별 볼륨 또는 전체 볼륨을 건너뛰고, 데이터베이스 백업 및 이미지를 건너뛸 수 있습니다.

특히 데이터베이스 백업을 건너뛰는 경우 백업은 SUSE Multi-Linux Manager 서비스를 중단하지 않고 생성되며, 2단계 백업 절차 중 1단계로 작동할 수 있습니다.

부분 백업은 데이터의 일부만을 고려하며, 백업되지 않았을 수 있는 다른 부분과의 잠재적인 종속성을 고려하지 않습니다. 따라서 백업/복원의 일관성을 보장할 수 없습니다.

절차: 데이터베이스 백업을 건너뛰어 부분 백업 만들기

1. 컨테이너 호스트에서 루트 권한으로 다음을 통해 백업:

```
mgradm backup create --skipdatabase $path
```

$path를 백업 위치의 경로로 바꿉니다.

절차: 볼륨을 건너뛰어 부분 백업 만들기

1. 컨테이너 호스트에서 루트 권한으로 다음을 통해 백업:

```
mgradm backup create --skipvolumes $volumes $path
```

$path를 백업 위치의 경로로 바꿉니다.

$volumes을 백업에서 제외할 볼륨 이름의 이름 또는 포함시킬 쉼표로 구분된 볼륨 목록으로 바꿉니다.

데이터베이스 볼륨을 제외한 모든 볼륨을 건너뛰려면 **all**을 사용합니다.

4.2.3. 추가 볼륨 백업

mgradm backup 명령은 SUSE Multi-Linux Manager 볼륨의 내부 목록을 사용합니다. 설치 중에 추가 볼륨을 구성했거나 백업에 추가 볼륨을 추가해야 하는 경우, **--extravolumes $volumes**를 사용하여 볼륨을 지정해야 합니다.

절차: 추가 사용자 지정 볼륨으로 백업 만들기

1. 컨테이너 호스트에서 루트 권한으로 다음을 통해 백업:

```
mgradm backup create --extravolumes $volume $path
```

$path를 백업 위치의 경로로 바꿉니다.

$volumes를 백업에 포함시킬 볼륨 이름의 이름으로 바꾸거나 포함시킬 쉼표로 구분된 볼륨 목록으로 바꿉니다.

4.2.4. 수동 데이터베이스 백업 수행

절차: 수동 데이터베이스 백업 수행

1. 백업에 사용할 영구 저장소 공간을 할당합니다.
2. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 루트 권한으로 다음을 사용:

```
mgradm backup create --skipvolumes all --skipconfig --skipimages $path
```

4.3. 기존 백업에서 SUSE Multi-Linux Manager 복원

기존 백업에서 SUSE Multi-Linux Manager을(를) 복원하면 복원할 볼륨, 이미지, 구성에 대한 백업이 열거됩니다. 백업 생성의 경우와 달리 복원 작업에서는 내부 볼륨 목록을 사용하지 않고, 백업에 있는 모든 볼륨 또는 이미지를 자동으로 감지합니다.

복원할 항목 목록이 수집된 후에는 존재 여부 및 무결성 검사가 수행됩니다. 존재 여부 검사는 백업 복원이 실수로 기존 볼륨, 이미지 또는 구성을 덮어쓰지 않도록 보장합니다. 무결성 검사는 백업 항목 체크섬을 계산하여 수행됩니다.

두 검사가 모두 성공하면 실제 백업 복원이 수행됩니다.

SUSE Multi-Linux Manager 서비스는 백업 복원이 완료된 후에 자동으로 시작되지 않습니다.

절차: 기존 백업에서 복원

1. 컨테이너 호스트에서 루트 권한으로 다음을 사용하여 SUSE Multi-Linux Manager 서버 다시 배포:

```
mgradm stop
mgradm backup restore $path
mgradm start
```

$path를 백업 위치의 경로로 바꿉니다.

백업 확인은 시간이 오래 소요되는 작업일 수 있습니다. 다른 방법으로 백업 무결성이 보장되는 경우에는 **--skipverify**

옵션을 사용하여 확인을 건너뛸 수 있습니다.

어떤 이유로 백업에서 볼륨 복원을 건너뛰어야 하는 경우, **--skipvolumes $volumes** 옵션을 사용하면 됩니다.

4.3.1. 백업 복원 이후의 권장 단계

절차: SUSE Multi-Linux Manager 복원 이후의 권장 단계

1. SUSE Multi-Linux Manager Web UI를 사용하거나 컨테이너의 명령 프롬프트에서 **mgr-sync** 도구를 사용하여 SUSE Multi-Linux Manager 리포지토리를 다시 동기화하십시오. 제품을 다시 등록하거나 등록 및 SSL 인증서 생성 섹션을 건너뛸 수 있습니다.
2. 컨테이너 호스트에서 **/var/lib/containers/storage/volumes/var-spacewalk/_data/packages/**를 복원해야 하는지 확인합니다. **/var/lib/containers/storage/volumes/var-spacewalk/_data/packages/**가 백업에 없는 경우, 이를 복원해야 합니다. 소스 리포지토리를 사용할 수 있는 경우, 전체 채널 동기화를 통해 **/var/lib/containers/storage/volumes/var-spacewalk/_data/packages/**를 복원할 수 있습니다.

```
mgrctl exec -ti -- mgr-sync refresh --refresh-channels
```

3. **rhn-search** 서비스가 다음에 시작될 때 검색 인덱스의 재생성을 예약하십시오.

   이 명령어는 디버그 메시지만 생성하며, 오류 메시지는 생성하지 않습니다.

   컨테이너 호스트에서 다음을 입력합니다.

```
mgrctl exec -ti -- rhn-search cleanindex
```

4.4. Database Backup Management

This guide describes how to manage online database backups for the SUSE Multi-Linux Manager server using **mgradm**. The backup system is based on PostgreSQL continuous archiving Write-Ahead Logging (WAL), similarly like **smdba** in previous versions.

- To ensure the system can be fully recovered, perform regular backups of both the database and other system volumes.
- Regularly copy your backup volumes and WAL files to an external storage device or an off-site location for disaster recovery.

4.4.1. Overview

Continuous archiving reduces the risk of data loss by constantly backing up the transaction logs (WAL) to

a dedicated volume.

The backup system utilizes a dedicated Podman volume named **var-pgsql-walbackup** to store:

- A base backup (a full snapshot of the database).
- WAL segments (incremental changes since the last base backup).

Point-in-time recovery is currently not supported by **mgradm** and requires a manual workflow.

Only full recovery is supported by **mgradm** tool. The restore procedure will restore everything from the backup location.

4.4.2. 선행 조건

- **mgradm** tool installed on the host.
- SUSE Multi-Linux Manager server running in a Podman environment.

4.4.3. Enabling backups

To enable continuous archiving, run the following command:

```
mgradm backup db enable
```

This command:

- Verifies that the backup is not already enabled.
- Configures PostgreSQL (**postgresql.conf**) to enable **archive_mode** and sets the **archive_command**.
- Configures the **var-pgsql-walbackup** volume for the database service.
- Restarts the database service to apply changes.
- Performs an initial base backup inside the container.

If the backup is already configured but you need to re-initialize it, use the **--force** flag.

4.4.4. Checking backup status

You can verify the current state of the database backup system at any time:

```
mgradm backup db status
```

The command will report one of the following statuses:

- **enabled**: Backup is correctly configured and active.
- **disabled**: Continuous archiving is explicitly turned off.

- **misconfigured**: There is a discrepancy between the configuration and the runtime state (for example, missing volume, incorrect archive command).

4.4.5. Maintaining backups (rebasing)

Over time, the number of WAL segments can grow significantly, consuming disk space and increasing restoration time. It is recommended to periodically "rebase" the backup by creating a new base backup and starting a new WAL chain.

```
mgradm backup db rebase
```

This command creates a fresh full snapshot of the database without requiring a service restart.

Regularly rebase your database backups using command **mgradm backup db rebase** to minimize the time required for restore operations.

4.4.6. Restoring from backup

Restoring from a backup is a **destructive operation** that replaces the current database data with the content of the backup.

```
mgradm backup db restore
```

The restoration process:

- Stops the database service.
- Clears the current database data directory.
- Extracts the base backup from the backup volume.
- Configures PostgreSQL for recovery.
- Starts the database service.
- Replays the WAL segments to bring the database to the latest possible state.

The command will wait until the database has fully recovered and is ready to accept connections.

Ensure you have a recent base backup and all necessary WAL segments in the **var-pgsql-walbackup** volume before performing a restore.

4.4.7. Disabling backups

To turn off continuous archiving:

```
mgradm backup db disable
```

By default, this command disables archiving in the configuration and restarts the database but preserves the backup volume.

To also remove the backup volume and all stored backups, use:

```
mgradm backup db disable --purge-volume
```

Regularly test your restore procedures to verify data integrity and ensure you are prepared for a recovery.

4.4.8. External documentation

For more information on PostgreSQL, see PostgreSQL 16: Continuous Archiving and Point-in-Time Recovery (PITR)

Chapter 5. 채널 관리

채널은 소프트웨어 패키지를 그룹화하는 방법입니다.

SUSE Multi-Linux Manager에서 채널은 운영 체제 유형, 버전 및 아키텍처별로 그룹화된 기본 채널과 관련 기본 채널과 호환되는 하위 채널과 함께 기본 및 하위 채널로 그룹화됩니다. 클라이언트가 기본 채널에 할당되면 해당 시스템에서만 관련 하위 채널을 설치할 수 있습니다. 이러한 방식으로 채널을 구성하면 호환되는 패키지만 각 시스템에 설치됩니다.

소프트웨어 채널은 리포지토리를 사용하여 패키지를 제공합니다. 채널은 SUSE Multi-Linux Manager의 리포지토리를 미러링하고 패키지 이름 및 기타 데이터는 SUSE Multi-Linux Manager 데이터베이스에 저장됩니다. 채널과 연결된 리포지토리 수에는 제한이 없습니다. 그런 다음 클라이언트를 적절한 채널에 등록하여 해당 리포지토리의 소프트웨어를 클라이언트에 설치할 수 있습니다.

기본 채널 한 개에만 클라이언트를 할당할 수 있습니다. 그런 다음 해당 기본 채널 및 해당 하위 채널과 연결된 리포지토리에서 클라이언트가 패키지를 설치하거나 업데이트할 수 있습니다.

SUSE Multi-Linux Manager는 SUSE Multi-Linux Manager를 실행하기 위해 필요한 모든 사항을 제공하는 여러 벤더 채널을 제공합니다. SUSE Multi-Linux Manager 관리자 및 채널 관리자에게는 채널 관리 권한이 부여되며, 이를 통해 고유 사용자 정의 채널을 생성 및 관리할 수 있습니다. 환경에서 고유한 패키지를 사용하려면 사용자 정의 채널을 생성할 수 있습니다. 사용자 정의 채널을 기본 채널로 사용하거나 벤더 기본 채널과 연결할 수 있습니다.

사용자 정의 채널에 대한 자세한 내용은 **Administration › Custom-channels**에서 확인할 수 있습니다.

5.1. 채널 관리

기본적으로 모든 사용자는 시스템에 채널을 구독할 수 있습니다. Web UI를 사용하여 채널에 대한 제한 사항을 구현할 수 있습니다.

절차: 채널에 대한 구독자 액세스 제한

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 채널 목록**으로 이동하여 편집할 채널을 선택합니다.
2. **사용자별 구독 제한 사항** 섹션을 찾아 **조직 내에서 선택된 사용자만 이 채널 구독 가능**을 선택합니다. **[업데이트]**를 클릭하여 변경 사항을 저장합니다.
3. **구독자** 탭으로 이동하여 필요에 따라 사용자를 선택하거나 선택을 해제합니다.

5.2. 채널 삭제

명령 프롬프트에서 벤더 소프트웨어 채널을 삭제할 수 있습니다.

절차: 벤더 채널 삭제

1. SUSE Multi-Linux Manager 서버의 명령 프롬프트에서 루트 권한으로 사용 가능한 벤더 채널을 나열하고 삭제할 채널을 기록합니다.

```
mgr-sync list channels
```

2. 채널을 삭제합니다.

```
spacewalk-remove-channel -c <channel-name>
```

- 수동으로 채널을 삭제하는 것에 대한 자세한 내용은 **Administration › Removing-channels**에서 확인할 수 있습니다.
- 사용자 정의 채널 삭제에 대한 내용은 **Administration › Custom-channels**에서 확인할 수 있습니다.

5.3. 사용자 정의 채널

사용자 정의 채널을 사용하면 클라이언트를 업데이트하기 위해 사용할 수 있는 고유 소프트웨어 패키지 및 리포지토리를 생성할 수 있습니다. 또한, 사용자 환경에서 타사 벤더가 제공하는 소프트웨어도 사용할 수 있습니다.

이 섹션에서는 사용자 정의 채널을 생성, 관리 및 삭제하는 방법에 대해 자세히 설명합니다. 사용자 정의 채널을 생성 및 관리하려면 관리자 권한이 있어야 합니다.

5.3.1. Creating custom channels and repositories

사용자 정의 채널을 생성하기 전 연결할 기본 채널과 컨텐트에 사용할 리포지토리를 결정합니다.

클라이언트 시스템에 설치해야 하는 사용자 정의 소프트웨어 패키지가 있는 경우 사용자 정의 하위 채널을 생성하여 관리할 수 있습니다. SUSE Multi-Linux Manager Web UI에서 채널을 생성하고 패키지에 대한 리포지토리를 생성한 후 시스템에 채널을 할당해야 합니다.

클라이언트 시스템과 호환되지 않는 패키지가 포함된 하위 채널을 생성하지 마십시오.

벤더에서 제공하는 패키지를 사용하려면 벤더 채널을 기본 채널로 선택할 수 있습니다. 또는, **없음**을 선택하여 사용자 정의 채널을 기본 채널로 설정합니다.

Procedure: Creating a custom channel

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 채널**로 이동하고 **[채널 생성]**을 클릭합니다.
2. **소프트웨어 채널 생성** 페이지에서, 채널에 이름(예: **My Tools SLES 15 SP1 x86_64**) 및 레이블(예: **my-tools-sles15sp1-x86_64**)을 지정합니다. 레이블에는 공백 또는 대문자를 포함하지 않아야 합니다.
3. **상위 채널** 드롭다운에서 관련 기본 채널(예: **SLE-Product-SLES15-SP1-Pool for x86_64**)을 선택합니다. 패키지에 대해 호환되는 상위 채널을 선택했는지 확인합니다.
4. **아키텍처** 드롭다운에서 적절한 하드웨어 아키텍처(예: **x86_64**)를 선택합니다.
5. 필요한 대로, 연락처 세부 정보, 채널 액세스 제어 및 GPG 필드에 추가 정보를 입력합니다.
6. **[채널 생성]**을 클릭합니다.

사용자 정의 채널에는 추가 보안 설정이 필요한 경우가 있습니다. 많은 타사 벤더가 GPG로 패키지를 보호합니다. 사용자

정의 채널에서 GPG 보호 패키지를 사용하려면 메타데이터 서명에 사용된 GPG 키를 신뢰해야 합니다. 그런 다음 **서명된 메타데이터가 있습니까?** 확인란을 선택하여 신뢰할 수 있는 GPG 키와 패키지 메타데이터를 일치시킬 수 있습니다.

원격 채널 및 리포지토리가 GPG 키로 서명된 경우 해당 GPG 키를 임포트 및 신뢰할 수 있습니다. 예를 들어, SUSE Multi-Linux Manager 서버의 명령줄에서 **spacewalk-repo-sync** 실행:

```
/usr/bin/spacewalk-repo-sync -c <channellabelname> -t yum
```

이 명령과 절차는 임시 GPG 키 동기화에만 사용됩니다. 키를 영구적으로 저장하려면 이후의 이 섹션을 참조하십시오.

사용할 수 있는 경우 기본 **zypper** 호출이 키를 임포트합니다. Web UI는 이 기능을 제공하지 않습니다.

이 기능은 미러링할 리포지토리가 특수 방식으로 설정되고 서명 옆의 리포지토리에 "키"를 제공하는 경우에만 작동하며, OBS(Open Build Service)에 의해 생성된 모든 리포지토리에 해당합니다. 다른 리포지토리의 경우 아래 설명과 같은 특별한 준비 단계가 필요합니다.

기본적으로 새 채널을 생성할 때 **GPG 검사 활성화** 필드가 선택되어 있습니다. 채널에 사용자 정의 패키지 및 애플리케이션을 추가하려면 서명되지 않은 패키지를 설치할 수 있도록 이 필드의 선택을 취소해야 합니다. 신뢰할 수 없는 출처에서 제공된 패키지의 경우 GPG 검사를 비활성화하면 보안 위험이 발생할 수 있습니다.

You can only add a repository to the SUSE Multi-Linux Manager with the Web UI if it is a valid software repository. Check in advance that needed repository metadata are available. Tools such as **createrepo** and **reprepro** are useful in this regard. **mgrpush** can help with pushing a single RPM into a channel without creating a repository.

- For more information on **createrepo_c** see https://manpages.opensuse.org/Leap-15.6/createrepo_c/
- For more information on **reprepro** see https://manpages.opensuse.org/Leap-15.6/reprepro/

Procedure: Adding a software repository

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 리포지토리**로 이동하고 [**리포지토리 생성**]을 클릭합니다.
2. **리포지토리 생성** 페이지에서 저장소에 레이블(예: **my-tools-sles15sp1-x86_64-repo**)을 지정합니다.
3. **리포지토리 URL** 필드에 **repodata** 파일이 포함된 디렉토리의 경로(예: file:///opt/mytools/)를 입력합니다. 이 필드에 올바른 주소 지정 프로토콜을 사용할 수 있습니다.
4. **서명된 메타데이터가 있습니까?** 확인란의 선택을 취소합니다.
5. 선택 사항: 리포지토리에 클라이언트 인증서 인증이 필요한 경우 SSL 필드를 입력하십시오.

6. **[리포지토리 생성]**을 클릭합니다.

위의 절차는 미러링할 리포지토리가 서명 옆에 있는 리포지토리의 "키"를 제공하는 경우에만 작동합니다. OBS에 의해 생성된 모든 리포지토리의가 이에 해당하지만, 일반적으로 OBS에서 제공하지 않는 운영 체제의 다른 리포지토리는 해당하지 않습니다.

사용할 리포지토리에 GPG 키가 없는 경우 직접 제공한 후 수동으로 GPG 키를 키링으로 가져올 수 있습니다. **gpg** 명령줄 도구를 사용하여 키를 **/var/lib/spacewalk/gpgdir** 키링으로 가져오면 영구적으로 저장됩니다. 이 키는 chroot 환경이 제거되는 경우에도 유지됩니다.

Procedure: Creating a software repository with GPG key

1. 컨테이너 호스트에서 키를 키링으로 가져오는 명령은 다음과 같습니다.

```
mgradm gpg add /path/to/gpg.key
```

2. 자세한 내용은 **Client-configuration › Autoinst-owngpgkey**에서 확인할 수 있습니다.

uyuni_suite, uyuni_component 및 **uyuni_arch** 쿼리 파라미터를 사용하여 Debian 논플랫 리포지토리를 추가합니다.

uyuni_suite

필수입니다. Debian 설명서에서는 이를 **배포**라고도 합니다. 이 파라미터를 사용하여 적절한 소스를 지정합니다. 이 파라미터가 없으면 원래 접근 방식이 사용됩니다. 파라미터가 **/**로 끝나면 리포지토리가 플랫으로 식별됩니다.

uyuni_component

선택 사항입니다. 이 파라미터는 하나의 구성 요소만 지정할 수 있습니다. 구성 요소를 나열할 수는 없습니다. 적절한 소스 항목을 사용하면 여러 구성 요소를 지정할 수 있지만, Uyuni의 경우에는 불가능합니다. 대신 각 구성 요소에 대해 별도의 리포지토리를 생성해야 합니다.

uyuni_arch

선택 사항입니다. 생략하면 아키텍처 이름이 데이터베이스의 채널에 대한 SQL 쿼리로 계산됩니다. 채널의 아키텍처와 일치하지 않는 경우 **uyuni_arch**를 명시적으로 지정하십시오(아키텍처 이름이 모호한 경우도 있음).

예는 다음과 같습니다.

표 2. Debian non-flat repository mapping

유형	소스 라인 / URL
apt 소스 라인	**deb** https://pkg.jenkins.io/debian-stable **binary/**
URL 매핑	https://pkg.jenkins.io/debian-stable?uyuni_suite=binary/

유형	소스 라인 / URL
apt 소스 라인	**deb** https://deb.debian.org/debian/dists **stable main**
URL 매핑	https://deb.debian.org/debian/dists?uyuni_suite=stable&uyuni_component=main

다음 정보는 Debian 리포지토리 정의 형식에 대한 정보로 https://wiki.debian.org/DebianRepository/Format#Overview를 기반으로 합니다.

리포지토리 정의 형식은 다음과 같습니다.

```
deb uri suite [component1] [component2] [...]
```

예:

```
deb https://deb.debian.org/debian/dists stable main
```

또는

```
deb https://pkg.jenkins.io/debian-stable binary/
```

suite 및 **component**의 각 쌍에 대해 사양은 기본 URL **+ suite + component**에서 계산된 고유한 URL을 정의합니다.

Procedure: Assigning the repository to a channel

1. **소프트웨어 › 관리 › 채널**로 이동하여 새로 만든 사용자 정의 채널의 이름을 클릭하여 새 리포지토리를 사용자 정의 채널에 할당합니다.
2. **리포지토리** 탭으로 이동하여 채널에 할당할 리포지토리가 선택되어 있는지 확인합니다. **[리포지토리 저장]**을 클릭합니다.
3. 기본적으로 동기화 프로세스는 즉시 시작됩니다.

채널 동기화에 대한 자세한 내용은 administration:custom-channels.pdf에서 확인할 수 있습니다.

Procedure: Adding custom channels to an activation key

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 활성화 키**로 이동하여 사용자 정의 채널을 추가할 키를 선택합니다.

2. **세부 사항** 탭의 **하위 채널** 목록에서 연결할 채널을 선택합니다. 필요한 경우 여러 채널을 선택할 수 있습니다.
3. **[활성화 키 업데이트]**를 클릭합니다.

5.3.2. Custom channel synchronization

중요한 업데이트가 누락되지 않으려면 SUSE는 원격 리포지토리 변경 사항에 따라 사용자 정의 채널을 최신 상태로 유지할 것을 권장합니다.

기본적으로, 동기화는 생성한 모든 사용자 정의 채널에 대해 자동으로 수행됩니다. 특히, 다음의 경우에 수행됩니다.

- UI에서 또는 **spacewalk-common-channels**를 사용하여 채널에 리포지토리를 추가한 후
- 일일 작업 **mgr-sync-refresh-default**의 일부로, 이 경우 모든 사용자 정의 및 벤더 채널이 동기화됩니다.

이 기본 동작을 비활성화하려면, **/etc/rhn/rhn.conf**에 설정합니다.

```
java.unify_custom_channel_management = 0
```

이 속성을 끄면 동기화가 자동으로 수행되지 않으며, 사용자 정의 채널을 최신 상태로 유지하려면 다음을 수행합니다.

- **동기화** 탭으로 이동하고 **[지금 동기화]**를 클릭하여 수동으로 동기화합니다.
- **리포지토리** 탭에서 자동 동기화 일정을 설정합니다.

프로세스가 시작되면 채널이 동기화를 완료했는지 확인할 수 있는 다음과 같은 몇 가지 방법이 있습니다.

- SUSE Multi-Linux Manager Web UI에서 **관리 › 설치 마법사**로 이동하여 **제품** 탭을 선택합니다. 이 대화 상자는 동기화되는 중에 각 제품에 대한 진행률 막대를 표시합니다.
- SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 채널**로 이동한 다음, 리포지토리에 연결된 채널을 클릭합니다. **리포지토리 › 동기화** 탭으로 이동합니다. 리포지토리 이름 옆에 **동기화 상태**가 표시됩니다.
- 명령 프롬프트에서 동기화 로그 파일을 확인합니다.

  ```
  tail -f /var/log/rhn/reposync/<channel-label>.log
  ```

  각 하위 채널은 동기화 진행 중에 자체 로그를 생성합니다. 동기화가 완료되었는지 확인하려면 모든 기본 및 하위 채널 로그 파일을 확인해야 합니다.

다음 사용자 지정 채널 동기화 옵션을 사용할 수 있습니다.

리포지토리에서 제거된 채널의 패키지를 유지합니다.

이 작업을 수행하면 **strict** 모드가 꺼집니다.

오류를 동기화하지 않음

패치를 동기화하지 마십시오.

최신 패키지만 동기화

최신 패키지 버전만 동기화하십시오.

킥스타트 가능 트리 생성

이 옵션은 Kickstart 자동 설치에 사용할 수 있는 디렉토리 트리를 준비합니다.

오류 발생 시 종료

오류가 발생하면 동기화를 중지하십시오.

이러한 옵션은 각 채널에 대해 영구적으로 저장됩니다. **[지금 동기화]** 버튼도 채널 옵션을 저장한 후 동기화를 수행합니다.

5.3.3. Add packages and patches to custom channels

기존 채널에서 복제하지 않고 새 사용자 정의 채널을 생성하면 패키지 또는 패치가 포함되지 않습니다. SUSE Multi-Linux Manager Web UI를 사용하여 필요한 패키지 및 패치를 추가할 수 있습니다.

사용자 정의 채널은 복제되거나 사용자 정의된 패키지 또는 패치만 포함할 수 있으며 채널의 기본 아키텍처와 일치해야 합니다. 사용자 정의 채널에 추가된 패치는 채널에 있는 패키지에 적용되어야 합니다.

Procedure: Adding packages to custom channels

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 채널**로 이동하고 **패키지** 탭으로 이동합니다.
2. 선택 사항: **나열/제거** 탭으로 이동하여 현재 채널에 있는 모든 패키지를 확인합니다.
3. **추가** 탭으로 이동하여 채널에 새 패키지를 추가합니다.
4. 패키지를 제공할 상위 채널을 선택하고 **[패키지 보기]**를 클릭하여 목록을 채웁니다.
5. 사용자 정의 채널에 추가할 패키지를 확인하고 **[패키지 추가]**를 클릭합니다.
6. 선택 사항에 만족하면 **[추가 확인]**을 클릭하여 채널에 패키지를 추가합니다.
7. 선택 사항: **소프트웨어 › 관리 › 채널**로 이동하고 **패키지 › 비교** 탭으로 이동하여 현재 채널의 패키지를 다른 채널의 패키지와 비교할 수 있습니다. 두 채널을 동일하게 만들려면 **[차이 병합]** 버튼을 클릭하고 충돌을 해결합니다.

Procedure: Adding patches to a custom channel

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 채널**로 이동하고 **패치** 탭으로 이동합니다.
2. 선택 사항: **나열/제거** 탭으로 이동하여 현재 채널에 있는 모든 패치를 확인합니다.
3. **추가** 탭으로 이동하고 추가할 패치 종류를 선택하여 채널에 새 패치를 추가합니다.

4. 패치를 제공할 상위 채널을 선택하고 [관련 패치 보기]를 클릭하여 목록을 채웁니다.
5. 사용자 정의 채널에 추가할 패치를 확인하고 [확인]을 클릭합니다.
6. 선택 사항에 만족하면 [확인]을 클릭하여 채널에 패치를 추가합니다.

5.3.4. Manage custom channels

SUSE Multi-Linux Manager 관리자 및 채널 관리자는 모든 채널을 변경 또는 삭제할 수 있습니다.

다른 사용자에게 채널을 변경하거나 삭제할 수 있는 권한을 부여하려면 **소프트웨어 › 관리 › 채널**로 이동하여 편집할 채널을 선택합니다. **관리자** 탭으로 이동하여 사용자가 권한을 부여하는지 확인합니다. [업데이트]를 클릭하여 변경 사항을 저장합니다.

클라이언트 집합에 할당된 채널을 삭제하면 삭제된 채널 및 연결된 모든 클라이언트의 채널 상태가 즉시 업데이트됩니다. 이를 통해 변경 사항이 리포지토리 파일에 정확하게 반영될 수 있습니다.

Web UI를 사용하여 SUSE Multi-Linux Manager 채널을 삭제할 수 없습니다. 사용자 정의 채널만 삭제할 수 있습니다.

Procedure: Deleting custom channels

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 관리 › 채널**로 이동하여 삭제할 채널을 선택합니다.
2. [소프트웨어 채널 삭제]를 클릭합니다.
3. **채널 삭제** 페이지에서 삭제할 채널의 세부 정보를 확인하고 **시스템 가입 취소** 확인란을 선택하여 아직 가입 중인 시스템에서 사용자 정의 채널을 제거합니다.
4. [채널 삭제]를 클릭합니다.

채널이 삭제될 때 삭제된 채널의 일부에 해당하는 패키지는 자동으로 제거되지 않습니다. 채널이 삭제된 패키지는 업데이트할 수 없습니다.

SUSE Multi-Linux Manager Web UI에서 채널과 연결되지 않은 패키지를 삭제할 수 있습니다. **소프트웨어 › 관리 › 패키지**로 이동하여 제거할 패키지를 확인하고 [패키지 삭제]를 클릭합니다.

5.4. 타사 채널

SUSE Multi-Linux Manager에는 선택 사항 제3자 리포지토리의 콘텐츠를 해당 제품이 관리할 수 있는 일부 제품과 동기화하는 기능이 포함되어 있습니다.

일부 타사 GPG 키는 SUSE Multi-Linux Manager 데이터베이스에 기본적으로 포함되어 있지만, 포함되지 않은 GPG 키도 있습니다. 포함되지 않은 타사 키의 경우, 해당 타사 리포지토리를 동기화할 때 관련 키를 반드시 가져와야 합니다.

GPG 키를 가져오려면 다음 명령 구문을 사용하여 SUSE Multi-Linux Manager 호스트 서버에서 명령어를 실행합니다.

```
mgradm gpg add <path-to-gpg-key-file-or-URL>
```

1. 예제: 파일에서 GPG 키 추가:

```
mgradm gpg add repomd.xml.key
```

2. 예제: URL을 사용하여 원격 리포지토리에서 GPG 키 추가:

```
mgradm gpg add https://<3rd-party-domain>/path/to/repository/repodata/repomd.xml.key
```

SUSE Multi-Linux Manager GPG 데이터베이스에 현재 저장된 키를 나열하려면 다음 명령을 실행합니다.

```
mgrctl exec -ti -- gpg --homedir /var/lib/spacewalk/gpgdir --list-keys
```

클라이언트에 패키지를 설치할 때 클라이언트 자체에서도 GPG 키를 신뢰해야 할 수 있습니다. 이를 위해서는 해당 GPG 키가 클라이언트에도 있어야 합니다.

자세한 내용은 **Client-configuration › Gpg-keys**에서 확인할 수 있습니다.

5.5. 채널 제거

이 장에서는 수명이 종료된 제품을 정리하거나 공간을 확보하거나 기타 목적으로 SUSE Multi-Linux Manager에서 SUSE 제공 채널을 수동으로 제거하는 방법을 설명합니다.

5.5.1. 채널 제거 준비

채널을 제거하기 전에 제거할 채널의 레이블을 파악해야 합니다. 이 작업은 Web UI을(를) 사용하거나 명령줄에서 수행할 수 있습니다.

- 시스템이 현재 구독 중이거나 시스템을 구독할 예정인 채널을 제거하지 않도록 주의하십시오.
- 삭제할 채널에 현재 구독 중인 시스템이 있는 경우 해당 시스템을 업그레이드하거나 구독을 취소할 때까지 채널을 제거하지 않아야 합니다.

5.5.1.1. 채널 레이블 식별

절차: Web UI를 사용하여 채널 레이블 식별

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 채널 목록 › 전체**로 이동합니다.
2. 이 페이지에는 **채널 이름**이 표시됩니다. 채널 이름에 대한 링크를 선택하면 **채널 레이블** 필드가 표시됩니다.
3. 제거할 채널과 하위 채널을 식별합니다.

절차: 명령줄을 사용하여 채널 레이블 식별

1. SUSE Multi-Linux Manager 컨테이너 호스트에세 다음 명령을 실행하여 채널 목록을 확인할 수 있습니다.

```
mgrctl exec -ti -- spacewalk-remove-channel -l
```

5.5.1.2. 채널 시스템 구독 확인

절차: Web UI를 사용하여 채널 시스템 구독 확인

1. SUSE Multi-Linux Manager Web UI에서 **소프트웨어 › 채널 목록 › 전체**로 이동합니다.
2. 오른쪽에서 **시스템** 열을 찾습니다.
3. 제거할 채널 또는 하위 채널의 **시스템** 열을 확인합니다.

절차: 명령줄을 사용하여 채널 시스템 구독 확인

1. SUSE Multi-Linux Manager 컨테이너 호스트에서 다음 명령 실행:

```
mgrctl exec -ti -- 'spacecmd -- softwarechannel_listsystems <Channel Label>'
```

5.5.2. 채널 제거

메타데이터가 있는 채널은 명령줄 도구로만 제거할 수 있습니다. **spacewalk-remove-channel** 명령은 다른 채널에서 더 이상 참조하지 않는 경우 소프트웨어 패키지를 자동으로 제거합니다. 따라서 데이터베이스에서 메타데이터를 제거하고 스토리지 매체에서 파일을 제거합니다.

5.5.2.1. 하위 채널 제거

절차: 명령 줄을 사용한 하위 채널 제거

1. 개별 채널을 제거하려면 SUSE Multi-Linux Manager 컨테이너 호스트에서 다음 명령을 실행합니다.

```
mgrctl exec -ti -- spacewalk-remove-channel -c channel-label
```

2. 여러 채널을 동시에 제거하려면 각 채널에 **-c** 플래그를 사용한 다음 **channel-label**을 사용합니다. 예:

```
mgrctl exec -ti -- spacewalk-remove-channel -c channel-label1 -c channel-label2
```

5.5.2.2. 상위 및 모든 하위 채널 제거

절차: 상위 및 모든 하위 채널 제거

1. 모든 하위 채널이 있는 상위 채널을 제거하려면 **spacewalk-remove-channel**을 **-a** 옵션으로 실행하고 **parent-channel-label**을 **sles12-sp5-pool-x86_64**와 같은 채널 레이블로 바꿉니다.

```
mgrctl exec -ti -- spacewalk-remove-channel -a parent-channel-label
```

Chapter 6. 기밀 컴퓨팅

Confidential Computing는 데이터 무결성, 데이터 기밀성, 코드 무결성에 대한 보안 수준을 향상하는 환경의 일종인 하드웨어 기반 TEE(신뢰 실행 환경)를 사용하여 사용 중 데이터를 보호할 수 있는 기술입니다.

6.1. SUSE Multi-Linux Manager 포함 Confidential Computing

TEE의 신뢰성은 증명 프로세스를 통해 확인할 수 있습니다. SUSE Multi-Linux Manager은(는) 등록된 시스템의 인증 서버로 사용할 수 있습니다. 그리고 이 모드로 실행되는 시스템에 대한 보고서 페이지를 생성합니다. 이러한 시스템은 정기적으로 증명 및 검사해야 합니다. 과거 검사 기록도 저장되며 요청에 따라 사용할 수 있습니다.

Confidential Computing 증명은 사용되는 하드웨어와 증명 대상 시스템이 실행되는 환경에 따라 달라집니다.

Confidential Computing 증명은 x86_64 아키텍처에서만 사용할 수 있습니다.

6.2. 요구사항

Confidential Computing는 특성이 다음과 같은 환경에서 설정할 수 있습니다.

- 증명되는 시스템(가상 머신)이 SLES15 SP6이며 SUSE Multi-Linux Manager(으)로 부트스트랩됩니다.
- 하드웨어에 **AMD EPYC Milan CPU** 또는 **AMD EPYC Genoa CPU**가 있어야 합니다.
- Confidential Computing 증명을 허용하도록 BIOS를 구성해야 합니다.
- 호스트 OS 및 가상화 소프트웨어(KVM 및 libvirt)가 Confidential Computing를 지원해야 합니다.

6.3. 제한 사항

- SLES15 SP6에는 Confidential Computing 증명이 기술 미리 보기로 제공됩니다.
- SUSE Multi-Linux Manager에는 Confidential Computing 증명이 기술 미리 보기로 제공됩니다.
- 보안 부트가 증명되었습니다. 그러나 현재 KVM 보안 부트와 SNP 게스트는 함께 작동하지 않습니다.

6.4. SUSE Multi-Linux Manager에서 Confidential Computing 사용

호스트에서 Confidential Computing를 설정 및 구성하는 정확한 단계는 OS 제조업체 설명서를 참조하십시오.

절차: SUSE Multi-Linux Manager 설치 중에 증명 컨테이너 활성화

1. 증명 컨테이너는 **mgradm install podman**을 사용하여 SUSE Multi-Linux Manager을(를) 설치하는 동안 활성화됩니다.
2. 다음을 **mgradm.yaml** 파일에 추가합니다.

```
coco:
```

```
    replicas: 1
```

절차: SUSE Multi-Linux Manager 설치 후 증명 컨테이너 활성화

1. 설치 후 증명 컨테이너를 활성화하려면 명령줄 파라미터 **mgradm**을 사용합니다.
2. 명령 실행

```
mgradm scale uyuni-server-attestation --replicas 1
```

절차: SUSE Multi-Linux Manager 설치 후 증명 컨테이너 비활성화

1. 이미 활성화된 증명 컨테이너를 비활성화하려면 다음 명령을 실행합니다.

```
mgradm scale uyuni-server-attestation --replicas 0
```

절차: 증명 활성화

1. 선택한 시스템의 경우 탭 메뉴: 감사[기밀 컴퓨팅 > 설정]으로 이동합니다.
2. 토글 버튼을 선택하여 증명을 활성화합니다.
3. 드롭다운 목록의 **환경 유형** 필드에서 올바른 옵션을 선택합니다.
4. **[저장]** 버튼을 클릭하여 변경 내용을 저장합니다.

절차: 새 증명 예약

1. 선택한 시스템의 경우 tab **감사 › 기밀 컴퓨팅 › 증명 나열**로 이동합니다.
2. **[증명 예약]**을 클릭합니다. 새 양식이 열립니다.
3. **가장 먼저** 필드에서 증명 실행 시간을 선택합니다.
4. 필요한 경우 **추가 위치** 옵션을 선택하여 새로 생성한 증명을 작업 체인에 추가합니다.
5. **[일정]** 버튼을 클릭하여 새 증명 실행을 저장하고 예약합니다.

절차: 시스템의 증명 보고서 세부 정보 보기

1. 선택한 시스템의 경우 tab **감사 › 기밀 컴퓨팅 › 증명 나열**로 이동합니다.
2. 확인할 보고서를 찾아 선택합니다.
3. 선택한 증명 보고서 탭을 클릭하면 **Overview**가 열립니다.
4. 다음 탭 **SEV-SNP**로 이동합니다.
5. 마지막으로, 다음 탭 **Secure Boot**로 이동합니다.

절차: 감사에서 증명 보고서 보기

1. 탐색 모음에서 **감사 › 기밀 컴퓨팅**을 선택합니다.
2. 모든 증명의 목록이 기본 패널에 표시됩니다.

3. 확인할 보고서를 찾아 선택합니다.

6.4.1. 보고서 상태

증명 보고서의 상태는 다음 중 하나일 수 있습니다.

보류 중

예약된 증명의 기본 상태입니다. 프로세스가 아직 시작되거나 완료되지 않았기 때문에 보고서를 아직 사용할 수 없습니다.

성공

예약된 증명이 볼 수 있는 보고서를 생성하면 프로세스 상태는 **성공**입니다.

실패

예약된 증명이 실패하여 결과적으로 보고서가 생성되지 않으면 프로세스 상태는 **실패**입니다.

6.5. 관련 주제

Confidential Computing에 대한 자세한 내용은 여기를 참조하십시오.

Chapter 7. 컨텐트 라이프싸이클 관리

컨텐트 라이프싸이클 관리를 사용하면 패키지를 사용자 정의하고 테스트한 후 프로덕션 클라이언트를 업데이트할 수 있습니다. 제한된 유지 관리 기간 동안 업데이트를 적용해야 하는 경우 특히 유용합니다.

컨텐트 라이프싸이클 관리를 통해 소프트웨어 채널을 소스로 선택하고 환경에 맞게 조정하며 철저히 테스트를 수행한 후에 프로덕션 클라이언트에 설치할 수 있습니다.

벤더 채널을 직접 수정할 수는 없지만 복제한 후 패키지 및 사용자 정의 패치를 추가하거나 제거하여 복제를 수정할 수 있습니다. 이러한 복제된 채널을 테스트 클라이언트에 할당하여 예상대로 작동하는지 확인할 수 있습니다.

기본적으로 복제된 제조업체 채널은 원래 제조업체 채널과 일치하며 종속성을 자동으로 선택합니다. **/etc/rhn/rhn.conf**에 다음 옵션을 추가하여 복제된 채널에 대한 자동 선택을 비활성화할 수 있습니다.

```
java.cloned_channel_auto_selection = false
```

그런 다음 모든 테스트가 통과되면 복제된 채널을 운영 서버로 승격할 수 있습니다.

이는 소프트웨어 채널이 라이프싸이클 동안 이동할 수 있는 일련의 환경을 통해 달성됩니다. 대부분의 환경 라이프싸이클에는 최소한 테스트 및 프로덕션 환경이 포함되지만 필요한 만큼 많은 환경을 가질 수 있습니다.

이 섹션에서는 기본 컨텐트 라이프싸이클 절차와 사용 가능한 필터에 대해 설명합니다. 보다 구체적인 예는 **Administration › Content-lifecycle-examples**에서 확인할 수 있습니다.

7.1. 컨텐트 라이프싸이클 프로젝트 생성

컨텐트 라이프싸이클을 설정하려면 프로젝트부터 시작해야 합니다. 프로젝트는 소프트웨어 채널 소스, 패키지를 찾기 위해 사용되는 필터 및 빌드 환경을 정의합니다.

절차: 컨텐트 라이프싸이클 프로젝트 생성

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 프로젝트**로 이동하고 [**프로젝트 생성**]을 클릭합니다.
2. **레이블** 필드에 프로젝트의 레이블을 입력합니다. **레이블** 필드에는 소문자, 숫자, 마침표, 하이픈 및 밑줄만 사용할 수 있습니다.
3. **이름** 필드에 프로젝트를 설명하는 이름을 입력합니다.
4. [**생성**] 버튼을 클릭하여 프로젝트를 생성하고 프로젝트 페이지로 돌아갑니다.
5. [**소스 연결/해제**]를 클릭합니다.
6. **소스** 대화 상자에서 소스 유형을 선택하고 프로젝트의 기본 채널을 선택합니다. 선택한 기본 채널에 사용 가능한 하위 채널과 채널이 필수인지 권장인지에 대한 정보 등이 표시됩니다.

7. 필요한 하위 채널을 확인하고 [**저장**]을 클릭하여 프로젝트 페이지로 돌아갑니다. 이제 선택한 소프트웨어 채널이 표시되어야 합니다.
8. [**필터 연결/해제**]를 클릭합니다.
9. **필터** 대화 상자에서 프로젝트에 첨부할 필터를 선택합니다. 새 필터를 생성하려면 [**새 필터 생성**]을 클릭합니다.
10. [**환경 추가**]를 클릭합니다.
11. **환경 라이프싸이클** 대화 상자에서 첫 번째 환경에 이름, 레이블 및 설명을 지정하고 [**저장**]을 클릭합니다. **레이블** 필드에는 소문자, 숫자, 마침표, 하이픈 및 밑줄만 사용할 수 있습니다.
12. 라이프싸이클에 대한 모든 환경이 완료될 때까지 환경을 계속 생성합니다. 생성할 때 **앞에 삽입** 필드에서 환경을 선택하여 라이프싸이클에서 환경의 순서를 선택할 수 있습니다.

7.2. 필터 유형

SUSE Multi-Linux Manager를 사용하면 다양한 유형의 필터를 생성하여 프로젝트 빌드에 사용되는 컨텐트를 제어할 수 있습니다. 필터를 사용하면 빌드에 포함되거나 제외되는 패키지를 선택할 수 있습니다. 예를 들어, 모든 커널 패키지를 제외하거나 일부 패키지의 특정 릴리스만 포함할 수 있습니다.

지원되는 필터:

- 패키지 필터링
  - 이름 기준
  - 이름, 에포크, 버전, 릴리스 및 아키텍처 기준
  - 제공 이름 기준
- 패치 필터링
  - 권고 이름 기준
  - 권고 유형 기준
  - 시놉시스 기준
  - 키워드 기준
  - 날짜 기준
  - 영향을 받는 패키지 기준
- 모듈
  - 스트림 기준

컨텐트 필터링 중에는 패키지 종속성이 해결되지 않습니다.

필터와 함께 사용할 수 있는 여러 선택기가 있습니다. 사용 가능한 필터 유형은 선택한 필터 유형에 따라 다릅니다.

사용할 수 있는 선택기:

- 포함
- 일치(정규식 형식이어야 함)
- 같음
- 큼
- 크거나 같음
- 작거나 같음
- 작음
- 이상

7.2.1. 필터 규칙 파라미터

각 필터에는 **허용** 또는 **거부**로 설정할 수 있는 **규칙** 파라미터가 있습니다. 필터는 다음과 같이 처리됩니다.

- 패키지 또는 패치가 **거부** 필터를 충족하면 결과에서 제외됩니다.
- 패키지 또는 패치가 **허용** 필터를 만족하는 경우, 해당 항목은 결과에 포함됩니다(**거부** 필터에 의해 제외된 경우에도 동일). 패키지에 대한 **허용** 필터는 패키지 필터에서만 작동하고, 패치에 대한 **허용** 필터는 패치 필터에서만 작동합니다. 즉, 패키지 필터를 사용하면 패치 필터를 통해 필터링된 패키지를 다시 추가할 수 없고, 반대로 패치 필터를 통해 필터링된 패치를 패키지 필터를 통해 다시 추가할 수 없습니다.

이 동작은 일반 **거부** 필터를 사용하여 많은 수의 패키지 또는 패치를 제외하고 특정 **허용** 필터가 적용된 특정 패키지 또는 패치를 "선별"하려는 경우에 유용합니다.

컨텐트 필터는 조직 전체에 적용되며 프로젝트 간에 공유할 수 있습니다.

프로젝트에 이미 빌드된 소스가 포함되어 있는 경우 환경을 추가하면 기존 내용이 자동으로 채워집니다. 내용은 주기의 이전 환경이 있는 경우 해당 환경에서 가져옵니다. 이전 환경이 없으면 프로젝트 소스가 다시 빌드될 때까지 비어 있습니다.

7.3. 필터 템플릿

몇 가지 일반적인 시나리오에 대한 필터를 편리하게 생성할 수 있도록 SUSE Multi-Linux Manager는 필터 템플릿을 제공합니다. 이러한 템플릿을 적용하면 특정 사용 사례에 맞게 사전에 필터 집합을 만드는 데 유용합니다.

이 섹션에서는 사용할 수 있는 템플릿과 그 사용법에 대해 설명합니다.

7.3.1. SUSE 제품 기반 라이브 패치

라이브 패치가 포함된 프로젝트에서는 정기적인 향후 커널 패키지를 제외해야 라이브 패치 패키지만 클라이언트에 대한 업데이트로 제공됩니다. 반면에 이미 설치된 일반 커널 패키지는 시스템 무결성을 유지하기 위해 여전히 포함되어야 합니다.

이 템플릿을 적용하면 이 동작을 수행하기 위해 필요한 필터 세 개가 생성됩니다.

- 기본 커널 버전과 동일한 **kernel-default** 패키지를 포함하는 패치 허용
- **reboot_suggested** 키워드가 포함된 패치 거부
- **installhint(reboot-needed)**라는 이름을 제공하는 패키지가 포함된 패치 거부

라이브 패치 프로젝트를 설정하는 방법에 대한 자세한 내용은 administration:content-lifecycle-examples.pdf에서 확인할 수 있습니다.

절차: 템플릿 적용

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. 대화 상자에서 [**템플릿 사용**]을 클릭합니다. 그에 따라 입력이 변경됩니다.
3. **접두사** 필드에 이름 접두사를 입력합니다. 이 값은 템플릿에 의해 생성된 모든 개별 필터의 이름 앞에 추가됩니다. 템플릿이 프로젝트 컨텍스트에서 적용되는 경우 이 필드는 프로젝트 레이블로 미리 채워집니다.
4. **템플릿** 필드에서 **SUSE 제품 기반 라이브 패치**를 선택합니다.
5. **제품** 필드에서 라이브 패치를 설정할 제품을 선택합니다.
6. **커널** 필드의 선택한 제품에서 사용할 수 있는 버전 목록에서 커널 버전을 선택합니다. 이후의 일반 커널 패치를 거부하는 필터는 이 버전을 기반으로 합니다.
7. [**저장**]을 클릭하여 필터를 생성합니다.
8. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
9. [**필터 연결/해제**]를 클릭합니다.
10. 지정된 접두사가 있는 필터 세 개를 선택하고 [**저장**]을 클릭합니다.

7.3.2. 시스템 기반 라이브 패치

등록된 특정 시스템에 설치된 커널 버전을 기반으로 라이브 패치 프로젝트를 설정하려면 "시스템 기반 라이브 패치" 템플릿을 사용할 수 있습니다.

이 템플릿을 적용하면 이 동작을 수행하기 위해 필요한 필터 세 개가 생성됩니다.

- 기본 커널 버전과 동일한 **kernel-default** 패키지를 포함하는 패치 허용
- **reboot_suggested** 키워드가 포함된 패치 거부
- **installhint(reboot-needed)**라는 이름을 제공하는 패키지가 포함된 패치 거부

라이브 패치 프로젝트를 설정하는 방법에 대한 자세한 내용은 administration:content-lifecycle-examples.pdf에서 확인할 수 있습니다.

절차: 템플릿 적용

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. 대화 상자에서 [**템플릿 사용**]을 클릭합니다. 그에 따라 입력이 변경됩니다.
3. **접두사** 필드에 이름 접두사를 입력합니다. 이 값은 템플릿에 의해 생성된 모든 필터의 이름 앞에 추가됩니다. 템플릿이 프로젝트 컨텍스트에서 적용되는 경우 이 필드는 프로젝트 레이블로 미리 채워집니다.
4. **템플릿** 필드에서 **특정 시스템에 기반 라이브 패치**를 선택합니다.
5. **시스템** 필드에서, 목록에서 시스템을 선택하거나 시스템 이름을 입력하여 옵션 범위를 좁힙니다.
6. **커널** 필드에서 선택한 시스템에 설치된 버전 목록에서 커널 버전을 선택합니다. 이후의 일반 커널 패치를 거부하는 필터는 이 버전을 기반으로 합니다.
7. [**저장**]을 클릭하여 필터를 생성합니다.
8. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
9. [**필터 연결/해제**]를 클릭합니다.
10. 지정된 접두사가 있는 필터 세 개를 선택하고 [**저장**]을 클릭합니다.

7.3.3. 기본값이 있는 AppStream 모듈

프로젝트에 포함된 모듈형 리포지토리에서 모든 모듈을 사용할 수 있도록 하려면 이 필터 템플릿을 사용하여 자동으로 추가할 수 있습니다.

적용되면 이 템플릿은 모듈 및 기본 스트림별로 AppStream 필터를 생성합니다.

이 프로세스가 프로젝트 페이지에서 수행되면 필터가 프로젝트에 자동으로 추가됩니다. 그렇지 않으면 생성된 필터를 **컨텐트 라이프싸이클 › 필터**에 나열하고 필요에 따라 모든 프로젝트에 추가할 수도 있습니다.

각 개별 필터를 편집하여 다른 모듈 스트림을 선택하거나 완전히 제거하여 대상 리포지토리에서 해당 모듈을 제외할 수 있습니다.

모든 모듈 스트림이 서로 호환되는 것은 아니므로, 개별 스트림을 변경하면 모듈 종속성을 성공적으로 해결하지 못할 수 있습니다. 이 경우 프로젝트 세부 사항 페이지의 필터 창에 문제를 설명하는 오류가 표시되고 모든 모듈 선택이 호환될 때까지 빌드 버튼이 비활성화됩니다.

Red Hat Enterprise Linux 9 이상 모듈에는 정의된 기본 스트림이 없습니다. 그러므로 Red Hat Enterprise Linux 9개의 소스와 함께 이 템플릿을 사용해도 효과가 없습니다.

컨텐트 라이프싸이클 관리를 사용하여 AppStream 리포지토리를 설정하는 방법에 대한 자세한 내용은 administration:content-lifecycle-examples.pdf에서 확인할 수 있습니다.

절차: 템플릿 적용

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
2. **필터** 섹션에서 [**필터 연결/해제**]를 클릭한 후 [**새 필터 생성**]을 클릭합니다.
3. 대화 상자에서 [**템플릿 사용**]을 클릭합니다. 그에 따라 입력이 변경됩니다.
4. **접두사** 필드에 이름 접두사를 입력합니다. 이 값은 템플릿에 의해 생성된 모든 필터의 이름 앞에 추가됩니다. 템플릿이 프로젝트 컨텍스트에서 적용되는 경우 이 필드는 프로젝트 레이블로 미리 채워집니다.
5. **템플릿** 필드에서 **기본값 포함 AppStream 모듈**을 선택합니다.
6. **채널** 필드에서 모듈을 가져올 모듈형 채널을 선택합니다. 이 드롭다운에는 모듈형 채널만 표시됩니다.
7. [**저장**]을 클릭하여 필터를 생성합니다.
8. **필터** 섹션으로 스크롤하여 새로 연결된 AppStream 필터를 확인합니다.
9. 개별 필터를 편집/제거하여 필요한 대로 프로젝트를 조정할 수 있습니다.

7.4. 컨텐트 라이프싸이클 프로젝트 빌드

프로젝트를 생성하고 환경을 정의하며 소스와 필터를 첨부하면 처음으로 프로젝트를 빌드할 수 있습니다.

빌드는 연결된 소스에 필터를 적용하고 이를 프로젝트의 첫 번째 환경에 복제합니다.

동일한 벤더 채널을 여러 컨텐트 프로젝트의 소스로 사용할 수 있습니다. 이 경우 SUSE Multi-Linux Manager는 복제된 각 채널에 대해 새 패치 복제를 생성하지 않습니다. 대신 단일 패치 클론이 복제된 모든 채널 간에 공유됩니다. 벤더가 패치를 수정하면 문제가 발생할 수 있습니다. 예를 들어, 패치가 철회되거나 패치 내의 패키지가 변경된 경우가 이에 해당합니다. 컨텐트 프로젝트 중 하나를 빌드하면 채널이 컨텐트 프로젝트의 다른 환경이나 조직의 다른 컨텐트 프로젝트 채널에 있더라도 복제된 패치를 공유하는 모든 채널이 기본적으로 원본과 동기화됩니다. 조직 설정에서 자동 패치 동기화를 해제하여 이 동작을 변경할 수 있습니다. 나중에 패치를 공유하는 모든 채널에 대해 수동으로 패치를 동기화하려면 **소프트웨어 › 관리 › 채널**로 이동하고 동기화하려는 채널을 클릭한 후 **동기화** 하위 탭으로 이동합니다. 수동 패치 동기화도 패치를 공유하는 모든 조직 채널에 영향을 미칩니다.

절차: 컨텐트 라이프싸이클 프로젝트 빌드

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 프로젝트**로 이동하고 빌드할 프로젝트를 선택합니다.

   사용 가능한 환경이 있는지 확인한 후 프로젝트를 빌드하십시오.

2. 첨부된 소스 및 필터를 검토하고 [빌드]를 클릭합니다.
3. 이 빌드의 변경 또는 업데이트 사항을 설명하는 버전 메시지를 제공합니다.
4. **환경 라이프싸이클** 섹션에서 빌드 진행 상황을 모니터링할 수 있습니다.

빌드가 완료되면 환경 버전이 하나씩 증가하고 소프트웨어 채널과 같이 빌드된 소스를 클라이언트에 할당할 수 있습니다.

7.5. 환경 승격

프로젝트가 빌드되면 빌드된 소스를 순차적으로 환경으로 승격할 수 있습니다.

절차: 환경 승격

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 프로젝트**로 이동하고 작업할 프로젝트를 선택합니다.
2. **환경 라이프싸이클** 섹션에서 후속 환경으로 승격할 환경을 찾고 [승격]을 클릭합니다.
3. **환경 라이프싸이클** 섹션에서 빌드 진행 상황을 모니터링할 수 있습니다.

7.6. 환경에 클라이언트 할당

컨텐트 라이프싸이클 프로젝트를 빌드 및 승격할 때 SUSE Multi-Linux Manager는 소프트웨어 채널 트리를 생성합니다. 환경에 클라이언트를 추가하려면 클라이언트의 **시스템 세부 정보** 페이지의 **소프트웨어 › 소프트웨어 채널**을 사용하여 기본 및 하위 소프트웨어 채널을 클라이언트에 할당합니다.

새로 추가된 복제된 채널은 클라이언트에 자동으로 할당되지 않습니다. 소스를 추가 또는 승격하는 경우 채널 할당을 수동으로 확인하고 업데이트해야 합니다.

자동 할당은 향후 버전에서 SUSE Multi-Linux Manager에 추가될 예정입니다.

7.7. 컨텐트 라이프싸이클 관리 예제

이 섹션에는 컨텐트 라이프싸이클 관리를 사용하는 방법에 대한 몇 가지 일반적인 예가 포함되어 있습니다. 이러한 예를 사용하여 자신만의 개인화된 구현을 빌드하십시오.

7.7.1. 월간 패치 주기에 대한 프로젝트 생성

월간 패치 주기에 대한 예제 프로젝트는 다음으로 구성됩니다.

- **날짜 기준** 필터 생성
- 프로젝트에 필터 추가
- 새 프로젝트 빌드에 필터 적용
- 프로젝트에서 패치 제외
- 프로젝트에 패치 포함

7.7.1.1. 날짜 기준 필터 생성

날짜 기준 필터는 지정된 날짜 이후에 릴리스된 모든 패치를 제외합니다. 이 필터는 월간 패치 주기를 따르는 컨텐트 라이프싸이클 프로젝트에 유용합니다.

절차: 날짜 기준 필터 생성

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. **필터 이름** 필드에 필터 이름을 입력합니다. 예를 들어, **날짜를 기준으로 패치 제외**입니다.
3. **필터 유형** 필드에서 **패치(발급 날짜)**를 선택합니다.
4. **선택기** 필드에서 **이후 또는 같음**이 자동 선택됩니다.
5. 날짜 및 시간을 선택합니다.
6. [**저장**]을 클릭합니다.

7.7.1.2. 프로젝트에 필터 추가

절차: 프로젝트에 필터 추가

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 프로젝트**로 이동하고 목록에서 프로젝트를 선택합니다.
2. 사용 가능한 모든 필터를 보려면 [**필터 연결/해제**] 링크를 클릭합니다.
3. 새 **날짜 기준 패치 제외** 필터를 선택합니다.
4. [**저장**]을 클릭합니다.

7.7.1.3. 새 프로젝트 빌드에 필터 적용

새 필터가 필터 목록에 추가되지만, 프로젝트에 적용해야 합니다. 필터를 적용하려면 첫 번째 환경을 빌드해야 합니다.

절차: 필터 사용

1. [**빌드**]를 클릭하여 첫 번째 환경을 빌드합니다.
2. 선택 사항: 메시지를 추가합니다. 메시지를 사용하여 빌드 기록을 추적할 수 있습니다.
3. 테스트 서버에서 새 채널을 사용하여 필터가 올바르게 작동했는지 확인합니다.
4. [**승격**]을 클릭하여 컨텐트를 다음 환경으로 이동합니다. 필터 수가 많거나 매우 복잡한 경우 빌드 시간이 더 오래 걸립니다.

7.7.1.4. 프로젝트에서 패치 제외

테스트는 문제 발견에 도움이 될 수 있습니다. 문제가 발견되면 **날짜 기준** 필터 이전에 릴리스된 문제 패치를 제외하십시오.

절차: 패치 제외

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. **필터 이름** 필드에 필터 이름을 입력합니다. 예를 들어, **openjdk 패치 제외**입니다.
3. **필터 유형** 필드에서 **패치(권고 이름)**를 선택합니다.
4. **선택기** 필드에서 **같음**을 선택합니다.
5. **권고 이름** 필드에 권고의 이름을 입력합니다. 예를 들어, **SUSE-15-2019-1807**입니다.
6. [**저장**]을 클릭합니다.
7. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
8. [**필터 연결/해제**] 링크를 클릭한 후 **openjdk 패치 제외**를 선택하고 [**저장**]을 클릭합니다.

[**빌드**] 버튼으로 프로젝트를 다시 빌드하면 이전에 추가한 **날짜 기준** 필터와 함께 새 필터가 사용됩니다.

7.7.1.5. 프로젝트에 패치 포함

보안 경고를 받은 예시입니다. 현재 작업 중인 달의 하루가 지나고 며칠 후에 중요한 보안 패치가 릴리스되었습니다. 새 패치의 이름은 **SUSE-15-2019-2071**입니다. 이 새 패치를 환경에 포함해야 합니다.

허용 필터 규칙은 **거부** 필터 규칙의 제외 기능보다 우선합니다. 자세한 내용은 **Administration › Content-lifecycle**에서 확인할 수 있습니다.

절차: 프로젝트에 패치 포함

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. **필터 이름** 필드에 필터 이름을 입력합니다. 예를 들어, **커널 보안 수정 포함**입니다.
3. **필터 유형** 필드에서 **패치(권고 이름)**를 선택합니다.
4. **선택기** 필드에서 **같음**을 선택합니다.
5. **권고 이름** 필드에 **SUSE-15-2019-2071**을 입력하고 **허용**을 선택합니다.
6. [**저장**]을 클릭하여 필터를 저장합니다.
7. **컨텐트 라이프싸이클 › 프로젝트**로 이동하고 목록에서 프로젝트를 선택합니다.

8. [**필터 연결/해제**]를 클릭하고 **커널 보안 패치 포함**을 선택합니다.
9. [**저장**]을 클릭합니다.
10. [**빌드**]를 클릭하여 환경을 다시 빌드합니다.

7.7.2. 기존 월간 패치 주기 업데이트

월간 패치 주기가 완료되면 다음 달의 패치 주기를 업데이트할 수 있습니다.

절차: 월간 패치 주기 업데이트

1. **날짜 기준** 필드에서 필터 날짜를 다음 달로 변경합니다. 또는 새 필터를 생성하고 프로젝트에 대한 할당을 변경합니다.
2. **SUSE-15-2019-1807**에 대한 제외 필터를 프로젝트에서 연결 해제할 수 있는지 확인합니다. 이 문제를 해결하기 위해 사용할 수 있는 새 패치가 있을 수 있습니다.
3. 이전에 추가한 **allow** 필터의 연결을 해제합니다. 패치는 기본적으로 포함되어 있습니다.
4. 프로젝트를 다시 빌드하여 다음 달에 대한 패치가 포함된 새 환경을 생성합니다.

7.7.3. 라이브 패치로 프로젝트 향상

이 섹션에서는 라이브 패치를 위한 환경을 생성하기 위한 필터 설정에 대해 설명합니다.

라이브 패치 사용을 준비할 때 고려해야 하는 몇 가지 중요한 사항이 있습니다.

- 시스템에서 하나의 커널 버전만 사용해야 합니다. 라이브 패치 패키지는 특정 커널과 함께 설치됩니다.
- 라이브 패치 업데이트는 단일 패치로 제공됩니다.
- 새로운 라이브 패치 커널 시리즈를 시작하는 각 커널 패치에는 **재부팅 필요** 플래그가 표시됩니다. 이러한 커널 패치는 라이브 패치 도구와 함께 제공됩니다. 이러한 패치를 설치하면 다음 연도 이전에 시스템을 한 번 이상 재부팅해야 합니다.
- 설치된 커널 버전과 일치하는 라이브 패치 업데이트만 설치해야 합니다.
- 라이브 패치는 독립 실행형 패치로 제공됩니다. 현재 설치된 커널 버전보다 높은 커널 버전이 포함된 모든 일반 커널 패치는 제외되어야 합니다.

7.7.3.1. 상위 커널 버전 패키지 제외

이 예에서는 **SUSE-15-2019-1244** 패치로 시스템을 업데이트합니다. 이 패치에는 **kernel-default-4.12.14-150.17.1-x86_64**가 포함되어 있습니다.

kernel-default 및 **kernel-default-base**의 상위 버전이 포함된 모든 패치를 제외해야 합니다.

절차: 상위 커널 버전 패키지 제외

1. SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클 › 필터**로 이동하고 [**필터 생성**]을 클릭합니다.
2. **필터 이름** 필드에 필터 이름을 입력합니다. 예를 들어, **4.12.14-150.17.1 이상 커널 제외**입니다.
3. **필터 유형** 필드에서 **패치(패키지 포함)**를 선택합니다.
4. **선택기** 필드에서 **다음 버전 초과**를 선택합니다.
5. **패키지 이름** 필드에 **kernel-default**를 입력합니다.
6. Leave the **Epoch** field empty.
7. **버전** 필드에 **4.12.14**를 입력합니다.
8. **릴리스** 필드에 **150.17.1**을 입력합니다.
9. [**저장**]을 클릭하여 필터를 저장합니다.
10. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
11. [**필터 연결/해제**]를 클릭합니다.
12. **4.12.14-150.17.1 초과 커널 제외**를 선택하고 [**저장**]을 클릭합니다.

kernel-default-base 패키지에 대해서도 이 절차를 반복해야 합니다.

[**빌드**]를 클릭하면 새로운 환경이 생성됩니다. 새 환경에는 설치한 버전까지의 모든 커널 패치가 포함되어 있습니다.

더 높은 커널 버전이 포함된 모든 커널 패치가 제거됩니다. 라이브 패치 커널은 시리즈의 첫 번째 커널인 경우를 제외하고 계속 사용할 수 있습니다.

이 절차는 필터 템플릿을 사용하여 자동화할 수 있습니다. 라이브 패치 필터 템플릿을 적용하는 방법에 대한 자세한 내용은 administration:content-lifecycle.pdf에서 확인할 수 있습니다.

7.7.4. 라이브 패치를 위한 새 커널 버전으로 전환

특정 커널 버전에 대한 라이브 패치는 한 해만 사용할 수 있습니다. 한 해가 지나면 시스템의 커널을 업데이트해야 합니다. 다음 환경 변경을 실행합니다.

절차: 새 커널 버전으로 전환

1. 업그레이드할 커널 버전을 결정합니다. 예는 다음과 같습니다. **4.12.14-150.32.1**
2. 새 커널 버전 필터를 생성합니다.
3. 기존 커널과 연결된 모든 이전 라이브 패칭 필터를 분리합니다.

4. 새 커널 버전 필터를 연결합니다.
5. [**빌드**]를 클릭하여 환경을 다시 빌드합니다.

빌드가 완료되면 즉시 해당 필터 2개를 다시 연결해야 합니다. 이러한 필터들은 다음과 같습니다.

- **reboot_suggested** 키워드가 포함된 패치 거부
- **installhint(reboot-needed)**라는 이름을 제공하는 패키지가 포함된 패치 거부

새 환경에는 선택한 새 커널 버전까지의 모든 커널 패치가 포함되어 있습니다. 업그레이드한 이후에는 시스템을 재부팅해야 합니다. 새 커널은 일 년간 사용할 수 있습니다. 해당 연도에 설치된 모든 패키지는 현재 라이브 패치 커널 필터와 일치합니다.

7.7.5. AppStream 필터

컨텐트 라이프싸이클 관리 프로젝트에서 AppStream 필터를 사용하여 모듈형 리포지토리를 일반 리포지토리로 변환할 수 있습니다. 이 필터는 리포지토리에 패키지를 유지하고 모듈 메타데이터를 제거하여 이를 수행합니다. SUSE Multi-Linux Manager에서 결과 리포지토리는 일반 리포지토리와 동일한 방식으로 사용할 수 있습니다.

그러므로 이 프로세스는 AppStream 리포지토리 작업에서 반드시 필요한 프로세스가 아닙니다.

AppStream 필터는 대상 리포지토리에 포함할 단일 모듈 스트림을 선택합니다. 여러 필터를 추가하여 여러 모듈 스트림을 선택할 수 있습니다.

CLM 프로젝트에서 AppStream 필터를 사용하지 않으면 모듈형 소스의 모듈 메타데이터는 그대로 유지되고 대상 리포지토리에는 동일한 모듈 메타데이터가 포함됩니다. CLM 프로젝트에서 AppStream 필터가 한 개 이상 활성화되어 있으면 모든 대상 리포지토리가 일반 리포지토리로 변환됩니다.

일부 경우 모듈의 패키지를 포함하지 않고 일반 리포지토리를 구축하는 것이 가능합니다. 이를 위해서는 선택기 **none (disable moduleity)**를 사용하여 AppStream 필터를 추가하십시오. 그러면 대상 리포지토리의 모든 모듈이 비활성화됩니다. 이것은 Red Hat Enterprise Linux 9 클라이언트에 특히 유용합니다. 대부분의 모듈의 기본 버전은 AppStream 리포지토리에 일반 패키지로 이미 포함되어 있습니다.

AppStream 필터를 사용하려면 **Red Hat Enterprise Linux AppStreams**와 같은 모듈형 리포지토리가 있는 CLM 프로젝트가 필요합니다. 시작하기 전에 필요한 모듈을 소스로 포함했는지 확인합니다.

절차: AppStream 필터 사용

1. SUSE Multi-Linux Manager Web UI에서 Red Hat Enterprise Linux 8 또는 9 CLM 프로젝트로 이동합니다. 프로젝트에 AppStream 채널을 포함했는지 확인합니다.
2. [**필터 생성**]을 클릭하고 다음 파라미터를 사용합니다.

- **필터 이름** 필드에 새 필터의 이름을 입력합니다.
- **필터 유형** 필드에서 **모듈(스트림)**을 선택합니다.
- **선택기** 필드에서 **같음**을 선택합니다.
- **모듈 이름** 필드에 모듈 이름을 입력합니다. 예를 들어, **postgresql**입니다.
- **스트림** 필드에 원하는 스트림의 이름을 입력합니다. 예를 들어, **10**입니다. 이 필드를 비워 두면 모듈의 기본 스트림이 선택됩니다.

3. 새 필터를 생성하려면 [**저장**]을 클릭합니다.
4. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
5. [**필터 연결/해제**]를 클릭하고 새 AppStream 필터를 선택한 후 [**저장**]을 클릭합니다.

필터 생성/편집 양식의 찾아보기 기능을 사용하여 모듈형 채널을 위해 사용할 수 있는 모듈 스트림 목록에서 모듈을 선택할 수 있습니다.

절차: 사용 가능한 모듈 스트림 찾아보기

1. SUSE Multi-Linux Manager Web UI에서 Red Hat Enterprise Linux 8 또는 9 CLM 프로젝트로 이동합니다. 프로젝트에 AppStream 채널을 포함했는지 확인합니다.
2. [**필터 생성**]을 클릭하고 다음 파라미터를 사용합니다.
   - **필터 이름** 필드에 새 필터의 이름을 입력합니다.
   - **필터 유형** 필드에서 **모듈(스트림)**을 선택합니다.
   - **선택기** 필드에서 **같음**을 선택합니다.
3. 모든 모듈형 채널을 보려면 **사용 가능한 모듈 찾아보기**를 클릭합니다.
4. 모듈 및 스트림을 탐색할 채널을 선택합니다.
   - **모듈 이름** 필드에서 검색할 모듈 이름을 입력하거나 목록에서 선택합니다.
   - **스트림** 필드에서 검색할 스트림 이름을 입력하거나 목록에서 선택합니다.

채널 선택은 모듈 탐색을 위해서만 사용할 수 있습니다. 선택한 채널은 필터와 함께 저장되지 않으며 어떠한 방식으로도 CLM 프로세스에 영향을 미치지 않습니다.

대상 리포지토리에 포함할 다른 모듈 스트림에 대한 추가 AppStream 필터를 생성할 수 있습니다. 선택한 스트림이 활용하는 모든 모듈 스트림이 자동으로 포함됩니다.

충돌하거나 호환되지 않거나 누락된 모듈 스트림을 지정하지 않도록 주의하십시오. 예를 들어, 동일한 모듈에서 스트림을 두 개 선택하는 것은 적절하지 않습니다.

절차: 모듈성 비활성화

1. SUSE Multi-Linux Manager Web UI에서 Red Hat Enterprise Linux 8 또는 9 CLM 프로젝트로 이동합니다. 프로젝트에 AppStream 채널을 포함했는지 확인합니다.
2. **[필터 생성]**을 클릭하고 다음 파라미터를 사용합니다.
   - **필터 이름** 필드에 새 필터의 이름을 입력합니다.
   - **필터 유형** 필드에서 **모듈(스트림)**을 선택합니다.
   - **선택기** 필드에서 **없음(모듈성 비활성화)**을 선택합니다.
3. 새 필터를 생성하려면 **[저장]**을 클릭합니다.
4. **컨텐트 라이프싸이클 › 프로젝트**로 이동하여 프로젝트를 선택합니다.
5. **[필터 연결/해제]**를 클릭하고 새 AppStream 필터를 선택한 후 **[저장]**을 클릭합니다.

이 작업을 수행하면 대상 리포지토리에서 모듈 메타데이터가 효과적으로 제거됩니다. 단, 모듈에 속한 패키지는 제외입니다.

Web UI에서 **[빌드]** 버튼을 사용하여 CLM 프로젝트를 빌드할 때 대상 리포지토리는 선택한 모듈 스트림의 패키지를 포함하는 모듈이 포함되지 않은 일반 리포지토리입니다.

Red Hat Enterprise Linux 8 프로젝트에서 모듈성을 모두 비활성화하면 Red Hat Enterprise Linux 8이 정상적으로 작동하려면 일부 모듈이 필수적이므로 잘못된 환경이 구성할 수 있습니다.

Chapter 8. 컨텐트 스테이징

스테이징은 클라이언트가 패키지를 설치하기 전 미리 다운로드하기 위한 용도로 사용됩니다. 그러면 예약 즉시 패키지를 설치할 수 있으므로 유지 관리 기간에 드는 시간을 줄일 수 있습니다.

8.1. 컨텐트 스테이징 활성화

조직 전체에서 컨텐트 스테이징을 관리할 수 있습니다. SUSE Multi-Linux Manager Web UI에서 **관리자 › 조직**으로 이동하여 사용 가능한 조직 목록을 확인합니다. 조직 이름을 클릭하고 **컨텐트 스테이징 활성화** 상자를 선택하여 이 조직의 클라이언트가 패키지 데이터를 준비할 수 있도록 합니다.

조직을 만들고 관리하려면 SUSE Multi-Linux Manager 관리자로 로그인해야 합니다.

명령 프롬프트에서 스테이징을 활성화하려면 **/etc/sysconfig/rhn/up2date**를 편집하고 다음 행을 추가 또는 편집:

```
stagingContent=1
stagingContentWindow=24
```

stagingContentWindow 파라미터는 시간으로 표시되는 시간 값으로 다운로드 시작 시점을 결정하며, 예약된 설치 또는 업데이트 시간 이전의 시간입니다. 이 예에서 컨텐트는 설치 24시간 전에 다운로드됩니다. 다운로드 시작 시간은 시스템에서 선택한 연락 방법에 따라 다릅니다.

다음에 작업이 예약되면 패키지가 자동으로 다운로드되지만 설치되지는 않습니다. 예약된 시간에 준비된 패키지가 설치됩니다.

8.2. 컨텐트 스테이징 구성

컨텐트 스테이징을 구성하는 데 사용되는 파라미터 두 개:

- **salt_content_staging_advance**는 컨텐트 준비 기간을 시작하는 사전 시간(시간)으로, 패키지를 다운로드할 수 있는 설치 시작 전 시간입니다.
- **salt_content_staging_window**는 컨텐트 준비 기간(시간)으로, 클라이언트가 설치를 시작하기 전에 패키지를 준비하는 데 드는 시간입니다.

예를 들어, **salt_content_staging_advance**가 6시간으로 설정되고 **salt_content_staging_window**가 2시간으로 설정되면 스테이징 기간은 설치 여섯 시간 전에 시작되어 두 시간 동안 열려 있습니다. 설치가 시작될 때까지 남은 네 시간 동안 패키지가 다운로드되지 않습니다.

salt_content_staging_advance 및 **salt_content_staging_window** 패키지에 동일한 값을 설정하면 설치가 시작될 때까지 패키지를 다운로드할 수 있습니다.

/usr/share/rhn/config-defaults/rhn_java.conf에서 컨텐트 스테이징 파라미터를 구성합니다.

다음은 기본값입니다.

- **salt_content_staging_advance: 8 hours**
- **salt_content_staging_window: 8 hours**

이러한 파라미터가 올바르게 작동하려면 컨텐트 스테이징을 활성화해야 합니다.

Chapter 9. 연결이 해제된 설정

SUSE Multi-Linux Manager를 인터넷에 연결할 수 없는 경우 연결이 해제된 환경 내에서 사용할 수 있습니다.

리포지토리 미러링 도구(RMT)는 SUSE Linux Enterprise 15 이상에서 사용할 수 있습니다. RMT는 이전 SUSE Linux Enterprise 설치에서 사용할 수 있는 가입 관리 도구(SMT)를 대체합니다.

연결이 해제된 SUSE Multi-Linux Manager 설정에서 RMT 또는 SMT는 외부 네트워크를 사용하여 SUSE Customer Center에 연결합니다. 모든 소프트웨어 채널과 리포지토리는 이동식 스토리지 장치에 동기화됩니다. 그런 다음 스토리지 장치를 사용하여 연결이 해제된 SUSE Multi-Linux Manager 설치를 업데이트할 수 있습니다.

이 설정을 사용하면 SUSE Multi-Linux Manager 설치를 연결이 해제된 오프라인 환경에서 유지될 수 있습니다.

SUSE Multi-Linux Manager 서버를 직접 관리하려면 RMT 또는 SMT 인스턴스를 사용해야 합니다. 이는 캐스케이드에서 두 번째 RMT 또는 SMT 인스턴스를 관리하기 위해 사용할 수 없습니다.

RMT에 대한 자세한 내용은 https://documentation.suse.com/sles/15-SP7/html/SLES-all/book-rmt.html에서 확인할 수 있습니다.

9.1. SCC에서 채널 및 리포지토리 동기화

9.1.1. RMT 동기화

SUSE Linux Enterprise 15 설치에서 RMT를 사용하여 SUSE Linux Enterprise 12 이상을 실행하는 클라이언트를 관리할 수 있습니다.

각 SUSE Multi-Linux Manager 설치에 대해 전용 RMT 인스턴스를 설정하는 것이 좋습니다.

절차: RMT 설정

1. RMT 인스턴스에서 다음 RMT 패키지를 설치합니다.

```
zypper in rmt-server
```

2. YaST를 사용하여 RMT를 구성합니다.

```
yast2 rmt
```

3. 프롬프트에 따라 설치를 완료합니다.

RMT 설정에 대한 자세한 내용은 https://documentation.suse.com/sles/15-SP7/html/SLES-all/book-rmt.html에서 확인할 수 있습니다.

절차: RMT와 SCC 동기화

1. RMT 인스턴스에서 조직에 대해 사용할 수 있는 모든 제품 및 리포지토리 나열:

```
rmt-cli products list --all
rmt-cli repos list --all
```

2. 조직에 대해 사용할 수 있는 모든 업데이트를 동기화합니다.

```
rmt-cli sync
```

systemd를 사용하여 정기적으로 동기화하도록 RMT를 구성할 수도 있습니다.

3. 필요한 제품을 활성화합니다. 예를 들어, SLES 15를 동기화하려면 다음을 수행해야 합니다.

```
rmt-cli product enable sles/15/x86_64
```

4. 동기화된 데이터를 이동식 스토리지로 엑스포트합니다. 이 예에서 스토리지 매체의 마운트 경로 **/mnt/usb**:

```
rmt-cli export data /mnt/usb
```

5. 활성화된 리포지토리를 이동식 스토리지로 엑스포트합니다.

```
rmt-cli export settings /mnt/usb
rmt-cli export repos /mnt/usb
```

RMT 사용자가 쓸 수 있는 디렉토리에 외부 스토리지가 마운트되었는지 확인합니다. **/etc/rmt.conf**의 **cli** 섹션에서 RMT 사용자 설정을 변경할 수 있습니다.

9.1.2. SMT 동기화

SMT는 SUSE Linux Enterprise 12에 포함되어 있으며 SUSE Linux Enterprise 10 이상을 실행하는 클라이언트를 관리하기 위한 용도로 사용할 수 있습니다.

SMT를 사용하려면 리포지토리와 패키지를 동기화하기 위해 SMT 인스턴스에 로컬 미러 디렉토리를 생성해야 합니다.

SMT 설치 및 구성에 대한 자세한 내용은 https://documentation.suse.com/sles/12-SP5/html/SLES-all/book-smt.html에서 확인할 수 있습니다.

절차: SMT와 SCC 동기화

1. SMT 인스턴스에서 데이터베이스 대체 파일을 생성합니다.

```
smt-sync --createdbreplacementfile /tmp/dbrepl.xml
```

2. 동기화된 데이터를 이동식 스토리지로 엑스포트합니다. 이 예제에서는 스토리지 매체가 **/mnt/usb**에 마운트되어 있습니다.

```
smt-sync --todir /mnt/usb
smt-mirror --dbreplfile /tmp/dbrepl.xml --directory /mnt/usb \
           --fromlocalsmt -L /var/log/smt/smt-mirror-export.log
```

```
curl https://scc.suse.com/multi-linux-manager/product_tree.json -o
/mnt/usb/product_tree.json
```

RMT 사용자가 쓸 수 있는 디렉토리에 외부 저장소가 마운트되었는지 확인합니다. **/etc/smt.conf**에서 SMT 사용자 설정을 변경할 수 있습니다.

9.2. 필수 채널

SUSE Multi-Linux Manager가 지정된 채널을 동기화하려면 해당 SUSE Multi-Linux Manager 클라이언트 도구 채널이 필요합니다. 이러한 채널이 활성화되어 있지 않으면 SUSE Multi-Linux Manager에서 해당 제품을 감지하지 못할 수 있습니다.

다음 명령을 실행하여 해당 필수 채널을 활성화합니다.

SLES 12 및 SLES for SAP 또는 SLE HPC와 같은 제품 기반 제품

RMT: **rmt-cli products enable sle-manager-tools/12/x86_64**

SMT: **smt repos -p sle-manager-tools,12,x86_64**

SLES 15 및 SLES for SAP 또는 SLE HPC와 같은 제품 기반 제품

RMT: **rmt-cli products enable sle-manager-tools/15/x86_64**

SMT: **smt repos -p sle-manager-tools,15,x86_64**

그런 다음 채널을 미러링하고 엑스포트합니다.

기타 배포 또는 아키텍처를 활성화할 수 있습니다. 제품 채널 또는 리포지토리를 미러링하도록 활성화하는 방법에 대한 자세한 내용은 다음 설명서에서 확인할 수 있습니다.

RMT

https://documentation.suse.com/sles/15-SP7/html/SLES-all/cha-rmt-mirroring.html#sec-rmt-mirroring-enable-disable

SMT

https://documentation.suse.com/sles/12-SP5/single-html/SLES-smt/index.html#smt-mirroring-manage-domirror

9.3. 연결 해제된 서버

SUSE Multi-Linux Manager을(를) 연결 해제된 서버로 설정하려면 air-gapped 배포 지침을 따릅니다.

9.3.1. 배포

연결 해제된 서버는 공급자가 제공한 이미지를 사용하여 VM(Virtual Machine: 가상 머신)으로 배포하는 것이 좋습니다. SUSE Multi-Linux Manager Server의 air-gapped 배포에 대한 자세한 내용은 **Installation-and-upgrade › Container-deployment**에서 확인할 수 있습니다.

마지막 명령을 **--mirror** 옵션으로 실행하고 **</media/disk>**를 마운트 포인트로 바꿔야 합니다.

```
mgradm install podman --mirror </media/disk>
```

9.3.2. 동기화

SUSE Customer Center 데이터가 로드된 이동식 미디어를 사용하는 경우 이를 사용하여 연결이 끊긴 서버를 동기화할 수 있습니다.

동기화를 위해 사용하는 이동식 미디어는 항상 동일한 마운트 포인트에서 사용할 수 있어야 합니다. 저장소 매체가 마운트되지 않은 경우 동기화를 트리거하지 마십시오. 그러면 데이터가 손상됩니다.

절차: 연결이 끊긴 서버 동기화

1. Tomcat 서비스를 재시작합니다.

```
mgrctl exec -ti -- systemctl restart tomcat
```

2. 로컬 데이터를 새로 고침합니다.

```
mgrctl exec -ti -- mgr-sync refresh
```

3. 동기화를 실행합니다.

```
mgrctl exec -ti -- mgr-sync list channels
mgrctl exec -ti -- mgr-sync add channel channel-label
```

server.susemanager.fromdir을 설정하면 SUSE Multi-Linux Manager에서 SUSE Customer Center 인증서가 유효한지 여부를 확인할 수 없습니다. 대신, 경고 표시가 표시되고 SCC 온라인 확인이 수행되지 않습니다.

연결이 끊긴 설정의 대안으로 ISS(서버 간 동기화)를 사용하여 서버 간에 내용을 복사할 수 있습니다. 자세한 내용은 **Specialized-guides › Large-deployments**에서 확인할 수 있습니다.

Chapter 10. 디스크 공간 관리

디스크 공간이 부족하면 복구가 불가능한 SUSE Multi-Linux Manager 데이터베이스 및 파일 구조에 심각한 영향을 미칠 수 있습니다.

SUSE Multi-Linux Manager는 사용 가능한 디스크 공간을 위해 일부 디렉토리를 모니터링합니다. 모니터링되는 디렉토리 및 생성되는 경고를 수정할 수 있습니다. 모든 설정은 **/etc/rhn/rhn.conf** 구성 파일에서 구성됩니다.

모니터링되는 디렉토리 중 하나의 사용 가능 공간이 경고 임계값 미만으로 감소하면, 구성된 이메일 주소로 메시지가 전송되고 로그인 페이지 상단에 알림이 표시됩니다.

10.1. Monitored directories

SUSE Multi-Linux Manager에서 기본적으로 모니터링되는 디렉토리는 다음과 같습니다.

- **/var/spacewalk**
- **/var/cache**
- **/srv**

spacecheck_dirs 파라미터를 사용하여 모니터링할 디렉토리를 변경할 수 있습니다. 이 경우 공백으로 구분하여 여러 디렉토리를 지정할 수 있습니다.

예:

```
spacecheck_dirs = /var/spacewalk /var/cache /srv
```

볼륨에 대한 자세한 내용은 **Installation-and-upgrade › Container-management**에서 확인할 수 있습니다.

10.2. 임계값

SUSE Multi-Linux Manager는 기본적으로 모니터링되는 디렉토리에서 사용 가능한 총 공간이 10% 미만이면 경고를 생성합니다. 모니터링되는 디렉토리가 사용 가능한 공간의 5% 미만으로 감소하면 위험 경고를 생성합니다.

spacecheck_free_alert 및 **spacecheck_free_critical** 파라미터를 사용하여 이러한 경고 임계값을 변경할 수 있습니다.

예:

```
spacecheck_free_alert = 10
spacecheck_free_critical = 5
```

10.3. Shut down services

기본적으로 SUSE Multi-Linux Manager는 위험 경고 임계값에 도달하면 Spacewalk 서비스를 종료합니다.

spacecheck_shutdown 파라미터를 사용하여 이 동작을 변경할 수 있습니다. 값이 **true**이면 종료 기능이

활성화됩니다. 다른 값은 기능을 비활성화합니다.

예:

```
spacecheck_shutdown = true
```

10.4. Disable space checking

공간 검사 도구는 기본적으로 활성화되어 있습니다. 완전히 비활성화하기 위한 명령은 다음과 같습니다.

```
systemctl stop spacewalk-diskcheck.timer
systemctl disable spacewalk-diskcheck.timer
```

spacewalk-diskcheck.timer를 비활성화하면 경고 임계값에 도달한 경우 정기적인 이메일 경고가 중지되지만, 경고 알림은 로그인 페이지 상단에 계속 표시됩니다.

Chapter 11. 이미지 빌드 및 관리

11.1. 이미지 빌드 개요

SUSE Multi-Linux Manager을(를) 사용하면 시스템 관리자가 컨테이너와 OS 이미지를 빌드하고 결과를 이미지 저장소에 푸시할 수 있습니다.

절차: 이미지 빌드 및 푸시

1. 이미지 저장소를 정의합니다.
2. 이미지 프로파일을 정의하고 이를 소스(git 리포지토리 또는 디렉토리)와 연결합니다.
3. 이미지를 빌드합니다.
4. 이미지를 이미지 저장소로 푸시합니다.

SUSE Multi-Linux Manager은(는) Dockerfile과 Kiwi라는 두 가지 빌드 유형을 지원합니다. Kiwi 빌드 유형은 시스템, 가상 및 기타 이미지를 빌드하기 위한 용도로 사용됩니다.

Kiwi 빌드 유형의 이미지 저장소는 **srv-www** 볼륨에 파일 시스템 디렉토리로 미리 정의되어 있습니다.

해당 이미지 파일은 https://MANAGER-HOST/os-images/ORGANIZATION-ID/FILE-NAME에서 다운로드할 수 있습니다. 정확한 위치는 이미지 세부 정보 페이지에서 확인할 수 있습니다.

11.2. 컨테이너 이미지

11.2.1. 요구사항

컨테이너 기능은 SUSE Linux Enterprise Server 12 이상으로 구동되는 Salt 클라이언트에서 사용할 수 있습니다. 시작하기 전, 환경의 다음 요구 사항 충족 여부를 확인합니다.

- Dockerfile 및 구성 스크립트가 포함된 게시된 git 리포지토리입니다. 리포지토리는 공개 또는 비공개일 수 있으며, GitHub, GitLab 또는 BitBucket에서 호스팅되어야 합니다.
- Docker 레지스트리와 같이 올바르게 구성된 이미지 저장소입니다.

컨테이너에 대한 자세한 내용은 https://documentation.suse.com/container/all/html/Container-guide/에서 확인할 수 있습니다.

11.2.2. 빌드 호스트 생성

SUSE Multi-Linux Manager를 사용하여 이미지를 빌드하려면 빌드 호스트를 생성 및 구성해야 합니다. 컨테이너 빌드 호스트는 SUSE Linux Enterprise 12 이상을 실행하는 Salt 클라이언트입니다. 이 섹션에서는 빌드 호스트의 초기 구성에 대한 설명을 제공합니다.

빌드 호스트의 운영 체제는 대상 이미지의 운영 체제와 일치해야 합니다.

> 예를 들어, SUSE Linux Enterprise Server 15(SP2 이상) OS 버전으로 구동되는 빌드 호스트에서는 SUSE Linux Enterprise Server 15 기반 이미지를 빌드합니다. SUSE Linux Enterprise Server 12 SP4 또는 SUSE Linux Enterprise Server 12 SP4 OS 버전으로 구동되는 빌드 호스트에서는 SUSE Linux Enterprise Server 12 기반 이미지를 빌드합니다.
>
> 아키텍처 간 빌드는 지원되지 않습니다.

SUSE Multi-Linux Manager Web UI에서 다음 단계를 수행하여 빌드 호스트를 구성합니다.

절차: 호스트 빌드

1. **시스템 › 시스템** 개요 페이지에서 빌드 호스트로 지정할 Salt 클라이언트를 선택합니다.
2. 선택한 클라이언트의 **시스템 세부 사항** 페이지에서 컨테이너 모듈을 할당합니다. **소프트웨어 › 소프트웨어 채널**로 이동하여 컨테이너 모듈을 활성화(예: **SLE-Module-Containers15-Pool** 및 **SLE-Module-Containers15-Updates**)합니다. **[다음]**을 클릭하여 계속합니다.
3. **소프트웨어 채널 변경**을 예약하고 **[확인]**을 클릭합니다.
4. **시스템 세부 사항** 탭에서 **속성** 페이지를 선택하고, **추가 기능 시스템 유형** 목록에서 **컨테이너 빌드 호스트**를 활성화합니다. **[속성 업데이트]**를 클릭하여 확인합니다.
5. **Highstate**를 적용하여 모든 필수 패키지를 설치합니다. 시스템 세부 사항 탭에서 **상태 › Highstate**를 선택하고 **[Highstate 적용]**을 클릭합니다. 또는 SUSE Multi-Linux Manager 서버 명령줄에서 **Highstate**를 적용합니다.

```
salt '$your_client' state.highstate
```

11.2.3. 컨테이너에 대한 활성화 키 생성

SUSE Multi-Linux Manager를 사용하여 빌드한 컨테이너는 이미지를 빌드할 때 활성화 키와 연결된 채널을 리포지토리로 사용합니다. 이 섹션에서는 이러한 목적을 위해 임시 활성화 키를 생성하는 과정을 설명합니다.

> 컨테이너를 빌드하기 위해서는 **SUSE 관리자 기본값** 이외의 채널과 연결된 활성화 키가 필요합니다.

절차: 활성화 키 생성

1. **시스템 › 활성화 키**를 선택합니다.
2. **[키 생성]**을 클릭합니다.
3. **설명** 및 **키** 이름을 입력합니다. 드롭다운 메뉴를 사용하여 이 키와 연결할 **기본 채널**을 선택합니다.

4. [**활성화 키 생성**]으로 확인합니다.

자세한 내용은 **Client-configuration › Activation-keys**에서 확인할 수 있습니다.

11.2.4. 이미지 저장소 생성

빌드된 모든 이미지는 이미지 저장소로 푸시됩니다. 이 섹션에는 이미지 저장소 생성에 대한 정보가 포함되어 있습니다. 이미지 저장소는 일반적으로 레지스트리로 참조됩니다.

절차: 이미지 저장소 생성

1. **이미지 › 저장소**를 선택합니다.
2. **생성**을 클릭하여 새 저장소를 생성합니다.
3. **저장 유형**에서 올바른 유형을 선택합니다.
4. **레이블** 필드에 이미지 저장소의 이름을 정의합니다.
5. 컨테이너 레지스트리 호스트(내부 또는 외부)의 FQDN(정규화된 도메인 이름)으로 **URI** 필드를 입력하여 이미지 레지스트리 경로를 입력합니다.

   ```
   registry.example.com
   ```

   레지스트리 URI를 사용하여 이미 사용 중인 레지스트리의 이미지 저장소를 지정할 수도 있습니다.

   ```
   registry.example.com:5000/myregistry/myproject
   ```

6. [**생성**]을 클릭하여 새 이미지 저장소를 추가합니다.

11.2.5. 이미지 프로파일 생성

모든 컨테이너 이미지는 빌드 지침이 포함된 이미지 프로파일을 사용하여 빌드됩니다. 이 섹션에는 SUSE Multi-Linux Manager Web UI를 사용하여 이미지 프로파일을 생성하는 방법에 대한 정보가 포함되어 있습니다.

절차: 이미지 프로파일 생성

1. 이미지 프로파일을 생성하려면 **이미지 › 프로파일**을 선택하고 [**생성**]을 클릭합니다.
2. **레이블** 필드를 작성하여 이미지 프로파일의 이름을 입력합니다.

   컨테이너 이미지 태그의 형식이 **myproject/myimage**와 같은 경우 이미지 저장소 레지스트리 URI에 **/myproject** 접미사가 포함되어 있는지 확인합니다.

3. **이미지 유형**으로 **Dockerfile**을 사용합니다.
4. 드롭다운 메뉴를 사용하여 **대상 이미지 저장소** 필드에서 레지스트리를 선택합니다.
5. **경로** 필드에 GitHub, GitLab 또는 BitBucket 리포지토리 URL을 입력합니다. 경로는 빌드 호스트의 로컬 디렉토리일 수도 있습니다. URL은 **http**, **https** 또는 토큰 인증 URL이어야 합니다. GitHub 또는 GitLab의 경우 다음 형식 중 하나를 사용합니다.

   ### GitHub 경로 옵션

   - GitHub 단일 사용자 프로젝트 리포지토리

     ```
     https://github.com/USER/project.git#branchname:folder
     ```

   - GitHub 조직 프로젝트 리포지토리

     ```
     https://github.com/ORG/project.git#branchname:folder
     ```

   - GitHub 토큰 인증

     git 리포지토리가 비공개인 경우 인증을 포함하도록 프로파일의 URL을 수정합니다. 다음 URL 형식을 사용하여 GitHub 토큰으로 인증합니다.

     ```
     https://USER:<AUTHENTICATION_TOKEN>@github.com/USER/project.git#master:/con
     tainer/
     ```

   - GitLab 단일 사용자 프로젝트 리포지토리

     ```
     https://gitlab.example.com/USER/project.git#master:/container/
     ```

   - GitLab 그룹 프로젝트 리포지토리

     ```
     https://gitlab.example.com/GROUP/project.git#master:/container/
     ```

   - GitLab 토큰 인증

     git 리포지토리가 비공개이고 공개적으로 액세스할 수 없는 경우 인증을 포함하도록 프로파일의 git URL을 수정해야 합니다. 다음 URL 형식을 사용하여 GitLab 토큰으로 인증:

     ```
     https://gitlab-ci-
     token:<AUTHENTICATION_TOKEN>@gitlab.example.com/USER/project.git#master:/co
     ```

```
ntainer/
```

git 분기를 지정하지 않으면 기본적으로 **마스터** 분기가 사용됩니다. **폴더**가 지정되지 않은 경우 이미지 소스(Dockerfile 소스)는 GitHub 또는 GitLab 체크아웃의 루트 디렉토리에 있어야 합니다.

6. **활성화 키**를 선택합니다. 활성화 키를 통해 프로파일을 사용하는 이미지를 올바른 채널 및 패키지에 할당할 수 있습니다.

활성화 키를 이미지 프로파일과 연결하면 프로파일을 사용하는 모든 이미지가 올바른 소프트웨어 채널과 채널의 모든 패키지를 사용하도록 할 수 있습니다.

7. **[생성]** 버튼을 클릭합니다.

11.2.5.1. 예시 Dockerfile 소스

재사용할 수 있는 이미지 프로파일은 https://github.com/SUSE/manager-build-profiles에 게시됩니다.

ARG 파라미터를 사용하면 빌드된 이미지를 SUSE Multi-Linux Manager에서 제공하는 원하는 리포지토리와 연결할 수 있습니다. **ARG** 파라미터를 사용하면 SUSE Linux Enterprise Server의 이미지 버전을 빌드할 수도 있으며, 이 버전은 빌드 호스트 자체에서 사용하는 SUSE Linux Enterprise Server 버전과 다를 수 있습니다.

예를 들어 **ARG repo** 파라미터와 리포지토리 파일을 가리키는 **echo** 명령은 원하는 채널 버전에 대한 리포지토리 파일에 올바른 경로를 생성한 후 삽입합니다.

리포지토리는 이미지 프로파일에 할당한 활성화 키로 결정됩니다.

```
FROM registry.example.com/sles12sp2
MAINTAINER Tux Administrator "tux@example.com"

### 시작: 이러한 라인은 {productname}과(와) 함께 사용하기 위해 필요합니다.

ARG repo
ARG cert

# 올바른 인증서 추가
RUN echo "$cert" > /etc/pki/trust/anchors/RHN-ORG-TRUSTED-SSL-CERT.pem

# 인증서 신뢰 저장소 업데이트
RUN update-ca-certificates

# 이미지에 리포지토리 경로 추가
RUN echo "$repo" > /etc/zypp/repos.d/susemanager:dockerbuild.repo

### 끝: 이러한 라인은 {productname}과(와) 함께 사용하기 위해 필요합니다.

# 패키지 스크립트 추가
ADD add_packages.sh /root/add_packages.sh
```

```
# 패키지 스크립트 실행
RUN /root/add_packages.sh

# 빌드 후 이미지에서 리포지토리 경로 제거
RUN rm -f /etc/zypp/repos.d/susemanager:dockerbuild.repo
```

11.2.5.2. 사용자 정의 정보 키-값 쌍을 Docker buildargs로 사용

사용자 정의 정보 키-값 쌍을 할당하여 이미지 프로파일에 정보를 첨부할 수 있습니다. 또한, 이러한 키-값 쌍은 **buildargs**로 Docker 빌드 명령에 전달됩니다.

사용 가능한 사용자 정보 키 및 추가 키 생성에 대한 자세한 내용은 **Reference › Systems**에서 확인할 수 있습니다.

11.2.6. 이미지 빌드

이미지를 구축하는 방법은 두 가지입니다. 첫 번째 방법은 처음부터 새로 만드는 것입니다. 왼쪽 탐색 모음에서 **이미지 › 만들기**를 선택하거나 **이미지 › 프로파일** 목록에서 만들기 아이콘을 클릭한 다음 절차를 따릅니다.

절차: 이미지 빌드

1. **이미지 › 빌드**를 선택합니다.
2. 기본 **최신**이 아닌 다른 버전을 원하는 경우 다른 태그 이름을 추가하십시오(컨테이너에만 해당).
3. **빌드 프로파일** 및 **빌드 호스트**를 선택합니다.

   > 빌드 필드 오른쪽의 **프로파일 요약**을 확인합니다. 빌드 프로파일을 선택하면 선택한 프로파일에 대한 자세한 정보가 이 영역에 표시됩니다.

4. 빌드를 예약하려면 [**빌드**] 버튼을 클릭합니다.

11.2.7. 이미지 임포트

이미지를 가져오는 두 번째 방법은 임의의 이미지를 임포트한 후 검사하는 것입니다. 이를 위해서는, 왼쪽 탐색 모음에서 **이미지 › 이미지 목록**을 선택합니다. **임포트** 대화 상자의 텍스트 상자를 입력합니다. 처리가 완료되면 임포트한 이미지가 **이미지 목록** 페이지에 나열됩니다.

절차: 이미지 임포트

1. **이미지 › 이미지 목록**에서 [**임포트**]를 클릭하여 **이미지 임포트** 대화 상자를 엽니다.
2. **이미지 임포트** 대화 상자에서 다음 필드를 입력합니다.

   이미지 저장소

   검사를 위해 이미지를 끌어오는 레지스트리입니다.

이미지 이름

레지스트리에 있는 이미지의 이름입니다.

이미지 버전

레지스트리에 있는 이미지의 버전입니다.

빌드 호스트

이미지를 끌어오고 검사하는 빌드 호스트입니다.

활성화 키

이미지를 검사하는 소프트웨어 채널에 대한 경로를 제공하는 활성화 키입니다.

3. 확인하려면 **[임포트]**를 클릭합니다.

이미지 항목이 데이터베이스에 생성되고 SUSE Multi-Linux Manager에 대한 **이미지 검사** 작업이 예약됩니다.

처리가 완료되면 **이미지 목록**에서 임포트한 이미지를 찾을 수 있습니다. 이미지를 임포트했음을 나타내기 위해 **빌드** 열에 다른 아이콘이 표시됩니다. 임포트한 이미지의 상태 아이콘은 이미지의 **개요** 탭에서도 확인할 수 있습니다.

11.2.8. 문제 해결

11.2.8.1. 이미지 검사

기본 컨테이너 이미지(BCI)는 이 이미지를 실행하기 위한 모든 소프트웨어와 함께 제공되지만, BCI는 경량이므로 검사에 필요한 모든 도구와 라이브러리가 제공되지 않을 수 있습니다.

컨테이너 이미지를 검사할 때 다음과 같은 오류 메시지가 표시될 수 있습니다.

```
libssl.so.1.1:공유 객체 파일을 열 수 없음: 해당 파일 또는 디렉토리 없음
```

컨테이너 빌드 호스트와 Salt 번들을 검사에 사용하는 경우 뿐만 아니라 다른 상황에서도 BCI를 사용할 수 있지만, 검사를 작동시키기 위해서는 모든 필수 소프트웨어를 미리 추가해야 합니다.

이러한 문제를 방지하려면 **Dockerfile**을 사용하여 이미지에 **libopenssl**을 추가하고 이미지를 다시 빌드해야 합니다.

libexpat의 경우에도 마찬가지입니다.

11.2.8.2. 일반적인 문제

이미지 작업과 관련하여 알려진 몇 가지 문제는 다음과 같습니다.

- 레지스트리 또는 git 리포지토리에 액세스하기 위한 HTTPS 인증서는 사용자 정의 상태 파일을 사용하여 클라이언트에 배포해야 합니다.
- 현재 Docker를 사용한 SSH git 액세스는 지원되지 않습니다.

11.3. OS 이미지

OS 이미지는 Kiwi 빌드 시스템에 의해 빌드됩니다. 출력 이미지는 사용자 정의가 가능하며 PXE, QCOW2, LiveCD 또는 기타 유형의 이미지일 수 있습니다.

Kiwi 빌드 시스템에 대한 자세한 내용은 Kiwi 문서에서 확인할 수 있습니다.

11.3.1. 요구사항

The Kiwi image building feature is available for Salt clients running SUSE Linux Enterprise Server 15 and SUSE Linux Enterprise Server 12.

Kiwi 이미지 구성 파일 및 구성 스크립트는 다음 위치 중 한 곳에서 액세스할 수 있어야 합니다.

- Git 리포지토리
- HTTP 또는 HTTPS 호스팅 tar 아카이브
- 빌드 호스트 상의 로컬 디렉토리

git에서 제공하는 전체 Kiwi 리포지토리의 예는 https://github.com/SUSE/manager-build-profiles/tree/master/OSImage에서 확인할 수 있습니다.

Kiwi로 빌드된 OS 이미지를 실행하는 호스트에는 사용할 수 있는 1 GB 이상 RAM이 필요합니다. 디스크 공간은 이미지의 실제 크기에 따라 다릅니다. 자세한 내용은 기본 시스템 설명서에서 확인할 수 있습니다.

11.3.2. Accessing Git repositories via an HTTP/HTTPS proxy when building images

When a build host needs to fetch sources from a git repository that is only reachable through an HTTP/HTTPS proxy, you may see git timeouts during the build because the git client invoked by the Salt state does not always pick up system-wide proxy settings (for example **/etc/sysconfig/proxy** or environment variables).

To make git use the proxy for non-interactive builds create **/etc/gitconfig** with an http.proxy entry:

```
# /etc/gitconfig
[http]
    proxy = http://proxy.example.com:3128
# or for HTTPS
[http]
    proxy = https://proxy.example.com:3128
```

If the proxy requires authentication, prefer a credential helper or use a credential store instead of embedding credentials in the URL.

11.3.2.1. Automating with Salt

You can manage **/etc/gitconfig** on build hosts through Configuration Management, the same way as on any other managed system. Create a Salt state file and assign it to the build host via a config channel,

then apply a highstate from the SUSE Multi-Linux Manager UI.

If you want to source the proxy address from a System Custom Info field, use the **custom_info:** pillar prefix:

```
/etc/gitconfig:
  file.managed:
    - user: root
    - group: root
    - mode: '0644'
    - contents: |
        [http]
            proxy = {{ salt['pillar.get']('custom_info:build_server:git_proxy',
'http://proxy.example.com:3128') }}
```

11.3.3. Container-based Kiwi image build support

SUSE Multi-Linux Manager introduces a container-based Kiwi image build system, in addition to the existing legacy Kiwi and KiwiNG tools.

11.3.3.1. Configuration and overrides

Administrators can override the default behavior using the following pillar or custom values. To configure these, navigate to menu:[Systems>Custom System Info] in the Web UI and create the necessary keys.

The build system used depends on the underlying OS or specific pillar values:

- for SLE 11 / SLE 12: legacy Kiwi v7
- for SLE 15: KiwiNG (v9 and containerized Kiwi 10)

Administrators can override the default behavior using the following pillar or custom values:

- **use_kiwi_ng**: force the use of Kiwi 9,
- **use_kiwi_container**: force the use of containerized Kiwi 10. To enable this, set the value to **1**.

11.3.3.2. Version-specific configurations

When using a containerized build host for SUSE Linux Enterprise 15 profiles, specific configurations are required because SLES 15 profiles rely on Kiwi 9, while the default container behavior uses Kiwi 10.

To ensure the correct version is used for SLES 15 profiles, you must define the **kiwi_image** custom info key with the following value:

- **Key**: kiwi_image
- **Value**: **registry.suse.com/bci/kiwi:9**

If this key is not set, the system defaults to the latest version (e.g., **registry.suse.com/bci/kiwi:10.2**), which may result in build issues for SLES 15 profiles.

11.3.4. 빌드 호스트 생성

SUSE Multi-Linux Manager로 모든 종류의 이미지를 빌드하려면 빌드 호스트를 생성 및 구성합니다. OS 이미지 빌드 호스트는 SUSE Linux Enterprise Server 15(SP2 이상) 또는 SUSE Linux Enterprise Server 12(SP4 이상)에서 실행되는 Salt 클라이언트입니다.

이 절차는 빌드 호스트의 초기 구성을 설명합니다.

빌드 호스트의 운영 체제는 대상 이미지의 운영 체제와 일치해야 합니다.

예를 들어, SUSE Linux Enterprise Server 15(SP2 이상) OS 버전으로 구동되는 빌드 호스트에서는 SUSE Linux Enterprise Server 15 기반 이미지를 빌드합니다. SUSE Linux Enterprise Server 12 SP4 또는 SUSE Linux Enterprise Server 12 SP4 OS 버전으로 구동되는 빌드 호스트에서는 SUSE Linux Enterprise Server 12 기반 이미지를 빌드합니다.

아키텍처 간에는 빌드할 수 없습니다. 예를 들어, SUSE Linux Enterprise Server 15 SP3가 실행 중인 Raspberry PI(aarch64 아키텍처) 빌드 호스트에서는 Raspberry PI SUSE Linux Enterprise Server 15 SP3 이미지를 빌드해야 합니다.

절차: SUSE Multi-Linux Manager Web UI에서 빌드 호스트 구성

1. **시스템 › 개요** 페이지에서 빌드 호스트로 지정할 클라이언트를 선택합니다.
2. **시스템 세부 사항 › 속성** 탭으로 이동하고 **추가 기능 시스템 유형** > **OS 이미지 빌드 호스트** 상자를 선택합니다.
3. **[속성 업데이트]**로 확인합니다.
4. **시스템 세부 사항 › 소프트웨어 › 소프트웨어 채널**로 이동하고 빌드 호스트 버전에 따라 필요한 소프트웨어 채널을 활성화합니다.
   - SUSE Linux Enterprise Server 12 빌드 호스트에는 SUSE Multi-Linux Manager 클라이언트 도구(**SLE-Manager-Tools12-Pool** 및 **SLE-Manager-Tools12-Updates**)가 필요합니다.
   - SUSE Linux Enterprise Server 15 빌드 호스트에는 SUSE Linux Enterprise Server 모듈 **SLE-Module-DevTools15-SP2-Pool** 및 **SLE-Module-DevTools15-SP2-Updates**가 필요합니다.
   - 예약하고 **[확인]**을 클릭합니다.
5. **Highstate**를 적용하여 Kiwi 및 모든 필수 패키지를 설치합니다. 시스템 세부 사항 페이지에서 **상태 › Highstate**를 선택하고 **[Highstate 적용]**을 클릭합니다. 또는 SUSE Multi-Linux Manager 서버 명령줄에서 Highstate를 적용합니다.

```
salt '$your_client' state.highstate
```

11.3.4.1. SUSE Multi-Linux Manager 웹 서버 공인 인증서 RPM

빌드 호스트 프로비저닝은 SUSE Multi-Linux Manager 인증서 RPM을 빌드 호스트에 복사합니다. 이 인증서는 SUSE Multi-Linux Manager에서 제공하는 리포지토리에 액세스하기 위한 용도로 사용됩니다.

인증서는 **mgr-package-rpm-certificate-osimage** 패키지 스크립트에 의해 RPM으로 패키징됩니다. 패키지 스크립트는 새 SUSE Multi-Linux Manager 설치 중에 자동으로 호출됩니다.

spacewalk-certs-tools 패키지를 업그레이드하는 경우 업그레이드 시나리오는 기본값을 사용하여 패키지 스크립트를 호출합니다. 그러나 인증서 경로가 변경되었거나 사용할 수 없는 경우 업그레이드 절차가 완료된 후 **--ca-cert-full-path <path_to_certificate>**를 사용하여 패키지 스크립트를 수동으로 호출하십시오.

11.3.4.2. 패키지 스크립트 호출 예제

```
/usr/sbin/mgr-package-rpm-certificate-osimage --ca-cert-full-path /root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT
```

인증서가 포함된 RPM 패키지는 다음과 같이 Salt가 액세스할 수 있는 디렉토리에 저장됩니다.

```
/usr/share/susemanager/salt/images/rhn-org-trusted-ssl-cert-osimage-1.0-1.noarch.rpm
```

인증서가 포함된 RPM 패키지는 다음 로컬 빌드 호스트 리포지토리에 제공됩니다.

```
/var/lib/Kiwi/repo
```

빌드 소스에 SUSE Multi-Linux Manager SSL 인증서가 포함된 RPM 패키지를 지정하고 Kiwi 구성에 **부트스트랩** 섹션의 필수 패키지로 **rhn-org-trusted-ssl-cert-osimage**가 포함되어 있는지 확인합니다.

목록 3. config.xml

```
...
  <packages type="bootstrap">
    ...
    <package name="rhn-org-trusted-ssl-cert-osimage"
bootinclude="true"/>
  </packages>
...
```

11.3.5. OS 이미지에 대한 활성화 키 생성

이미지를 빌드할 때 OS 이미지가 리포지토리로 사용할 수 있는 채널과 연결된 활성화 키를 생성합니다.

활성화 키는 OS 이미지 빌드에 필수입니다.

OS 이미지를 빌드하려면 **기본** 활성화 키가 아닌 다른 채널과 연결된 활성화 키가 필요합니다.

절차: 활성화 키 생성

1. Web UI에서 **시스템 › 활성화 키**를 선택합니다.
2. **키 생성**을 클릭합니다.
3. **설명**, **키** 이름을 입력하고 드롭다운 상자를 사용하여 키와 연결할 **기본 채널**을 선택합니다.
4. **[활성화 키 생성]**으로 확인합니다.

자세한 내용은 **Client-configuration › Activation-keys**에서 확인할 수 있습니다.

11.3.6. 이미지 저장소 생성

OS 이미지에는 대용량 저장소가 필요할 수 있습니다. 기본적으로 이미지 저장소는 **srv-www** 볼륨을 사용합니다.

시스템, 가상 및 기타 이미지를 빌드하기 위해 사용되는 Kiwi 빌드 유형의 이미지 저장소는 아직 지원되지 않습니다.

해당 이미지 파일은 https://MANAGER-HOST/os-images/ORGANIZATION-ID/FILE-NAME에서 다운로드할 수 있습니다. 정확한 위치는 이미지 세부 정보 페이지에서 확인할 수 있습니다.

11.3.7. 이미지 프로파일 생성

Web UI를 사용하여 이미지 프로파일을 관리합니다.

절차: 이미지 프로파일 생성

1. 이미지 프로파일을 생성하려면 **이미지 › 프로파일**에서 선택하고 **[생성]**을 클릭합니다.
2. **레이블** 필드에 **이미지 프로파일**의 이름을 입력합니다.
3. **이미지 유형**으로 **Kiwi**를 사용합니다.
4. 이미지 저장소는 자동으로 선택됩니다.
5. Kiwi 구성 파일이 포함된 디렉토리에 **URL 구성**을 입력합니다. 예를 들어, https://github.com/SUSE/manager-build-profiles#master:OSImage/SLE-Micro54 같은 git URI를 입력합니다. 다른 옵션으로는 HTTP 또는 HTTPS로 호스팅되는 tar 아카이브 또는 빌드 호스트의 로컬 디렉토리가 있습니다. 자세한 내용은 이 섹션의 마지막에 제공되는 소스 형식 옵션을 참조하십시오.
6. 필요한 경우, **Kiwi 옵션**을 입력합니다. Kiwi 구성 파일이 여러 프로파일을 지정하는 경우 **--profile <name>**을 사용하여 활성 프로파일을 선택합니다. 기타 옵션은 Kiwi 설명서를 참조하십시오.

7. **활성화 키**를 선택합니다. 활성화 키를 사용하면 프로파일을 사용하는 이미지가 올바른 채널 및 패키지에 할당할 수 있습니다.

이미지 프로파일이 올바른 소프트웨어 채널 및 모든 패키지를 사용하도록 활성화 키를 이미지 프로파일과 연결합니다.

8. **[생성]** 버튼으로 확인합니다.

소스 형식 옵션

- 리포지토리에 대한 git/HTTP(S) URL

  빌드할 이미지의 소스가 포함된 공개 또는 비공개 git 리포지토리의 URL입니다. 리포지토리의 레이아웃에 따라 가능한 URL은 다음과 같습니다.

```
https://github.com/SUSE/manager-build-profiles
```

  URL에서 **#** 문자 뒤에 분기를 지정할 수 있습니다. 이 예에서는 다음 **master** 분기를 사용합니다.

```
https://github.com/SUSE/manager-build-profiles#master
```

  : 문자 뒤에 이미지 소스가 포함된 디렉토리를 지정할 수 있습니다. 이 예에서는 다음 **OSImage/POS_Image-JeOS6**을 사용합니다.

```
https://github.com/SUSE/manager-build-profiles#master:OSImage/POS_Image-JeOS6
```

- tar 아카이브에 대한 HTTP(S) URL

  웹 서버에서 호스팅되는 tar 아카이브(압축 또는 비압축)의 URL입니다.

```
https://myimagesourceserver.example.org/MyKiwiImage.tar.gz
```

- 빌드 호스트 상의 디렉토리 경로

  Kiwi 빌드 시스템 소스가 포함된 디렉토리 경로를 입력합니다. 이 디렉토리는 선택한 빌드 호스트에 위치해야 합니다.

```
/var/lib/Kiwi/MyKiwiImage
```

11.3.7.1. Kiwi 소스의 예제

Kiwi 소스는 최소한 **config.xml**로 구성됩니다. 일반적으로 **config.sh**와 **image.sh**도 함께 포함됩니다. 소스에는 **root** 하위 디렉토리의 최종 이미지에 설치할 파일도 포함될 수도 있습니다.

Kiwi 빌드 시스템에 대한 내용은 Kiwi documentation에서 확인할 수 있습니다.

SUSE는 SUSE/manager-build-profiles 공개 GitHub 리포지토리에서 완전하게 작동하는 이미지 소스의 예를 제공합니다.

목록 4. JeOS config.xml의 예

```
<?xml version="1.0" encoding="utf-8"?>

<image schemaversion="6.1" name="POS_Image_JeOS6">
    <description type="system">
        <author>Admin User</author>
        <contact>noemail@example.com</contact>
        <specification>SUSE Linux Enterprise 12 SP3 JeOS</specification>
    </description>
    <preferences>
        <version>6.0.0</version>
        <packagemanager>zypper</packagemanager>
        <bootsplash-theme>SLE</bootsplash-theme>
        <bootloader-theme>SLE</bootloader-theme>

        <locale>en_US</locale>
        <keytable>us.map.gz</keytable>
        <timezone>Europe/Berlin</timezone>
        <hwclock>utc</hwclock>

        <rpm-excludedocs>true</rpm-excludedocs>
        <type boot="saltboot/suse-SLES12" bootloader="grub2" checkprebuilt="true"
compressed="false" filesystem="ext3" fsmountoptions="acl" fsnocheck="true" image="pxe"
kernelcmdline="quiet"></type>
    </preferences>
    <!--    CUSTOM REPOSITORY
    <repository type="rpm-dir">
      <source path="this://repo"/>
    </repository>
    -->
    <packages type="image">
        <package name="patterns-sles-Minimal"/>
        <package name="aaa_base-extras"/> <!-- wouldn't be SUSE without that ;-) -->
        <package name="kernel-default"/>
        <package name="venv-salt-minion"/>
        ...
    </packages>
    <packages type="bootstrap">
        ...
        <package name="sles-release"/>
        <!-- this certificate package is required to access {productname} repositories
             and is provided by {productname} automatically -->
        <package name="rhn-org-trusted-ssl-cert-osimage" bootinclude="true"/>

    </packages>
    <packages type="delete">
        <package name="mtools"/>
        <package name="initviocons"/>
        ...
    </packages>
```

```
</image>
```

11.3.8. 이미지 빌드

Web UI를 사용하여 이미지를 빌드하거나 가져오는 방법은 2가지가 있습니다. **이미지 › 빌드**를 선택하거나 **이미지 › 프로파일** 목록에서 빌드 아이콘을 클릭합니다.

절차: 이미지 빌드

1. **이미지 › 빌드**를 선택합니다.
2. 기본 **최신**이 아닌 다른 버전을 원하는 경우 다른 태그 이름을 추가합니다(컨테이너에만 적용됨).
3. **이미지 프로파일** 및 **빌드 호스트**를 선택합니다.

   빌드 필드 오른쪽에 **프로파일 요약**이 표시됩니다. 빌드 프로파일을 선택하면 선택한 프로파일에 대한 자세한 정보가 여기에 표시됩니다.

4. 빌드를 예약하려면 **[빌드]** 버튼을 클릭합니다.

이미지 빌드 프로세스 중에 빌드 서버는 어떤 형태의 자동 마운터도 실행할 수 없습니다. 해당하는 경우, 루트 권한으로 실행 중인 Gnome 세션이 없는지 확인하십시오. 자동 마운터가 실행 중인 경우 이미지 빌드가 완료되지만, 이미지의 체크섬이 달라 실패하게 됩니다.

이미지 빌드가 성공하면 검사 단계가 시작됩니다. 검사 단계에서 SUSE Multi-Linux Manager는 이미지에 대한 정보를 수집합니다.

- 이미지에 설치된 패키지 목록
- 이미지의 체크섬
- 이미지 유형 및 기타 이미지 세부 사항

빌드된 이미지 유형이 **PXE**인 경우 Salt 열도 생성됩니다. 이미지 열은 데이터베이스에 저장되며 Salt 하위 시스템은 생성된 이미지에 대한 세부 사항에 액세스할 수 있습니다. 세부 사항에는 이미지 파일의 위치 및 제공 위치, 이미지 체크섬, 네트워크 부팅에 필요한 정보 등이 포함됩니다.

생성된 열은 연결된 모든 클라이언트에서 사용할 수 있습니다.

11.3.9. 문제 해결

이미지를 빌드하려면 여러 종속 단계가 필요합니다. 빌드가 실패하는 경우 Salt 상태 결과 및 빌드 로그를 조사하면 실패의 원인을 식별하는 데 도움이 될 수 있습니다. 빌드 실패 시 수행할 수 있는 검사:

- 빌드 호스트는 빌드 소스에 액세스할 수 있습니다.
- 빌드 호스트 및 SUSE Multi-Linux Manager 서버 모두에 이미지를 위한 충분한 디스크 공간이 있습니다.

- 활성화 키에 연결된 올바른 채널이 있습니다.
- 사용된 빌드 소스가 유효합니다.
- SUSE Multi-Linux Manager 공용 인증서가 포함된 RPM 패키지는 최신 버전이며 **/usr/share/susemanager/salt/images/rhn-org-trusted-ssl-cert-osimage-1.0-1.noarch.rpm**에서 사용할 수 있습니다. 공용 인증서 RPM을 새로 고치는 방법에 대한 자세한 내용은 빌드 호스트 생성에서 확인할 수 있습니다.

11.3.10. 제한 사항

이 섹션에는 이미지 관련 작업을 할 때 알려진 몇 가지 문제가 포함되어 있습니다.

- HTTP 소스 또는 git 리포지토리에 액세스하기 위해 사용되는 HTTPS 인증서는 사용자 정의 상태 파일을 통해 클라이언트에 배포하거나 수동으로 구성해야 합니다.
- Kiwi 기반 이미지 임포트는 지원되지 않습니다.

11.4. 빌드 이미지 목록

사용할 수 있는 빌드 이미지를 나열하려면 **이미지 › 이미지 목록**를 선택하십시오. 모든 이미지 목록이 표시됩니다.

이미지에 대해 표시되는 데이터에는 이미지 **이름**, **버전**, **리비전** 및 빌드 **상태**가 포함됩니다. 이미지에 사용할 수 있는 가능한 패치 및 패키지 업데이트 목록을 통해 이미지 업데이트 상태를 확인할 수도 있습니다.

OS 이미지의 경우 **이름** 및 **버전** 필드는 Kiwi 소스에서 제공되며 빌드가 성공적으로 완료되면 업데이트됩니다. 빌드 중 또는 빌드 실패 후, 이 필드에는 프로파일 이름을 기반으로 하는 임시 이름이 표시됩니다.

리비전은 빌드가 성공할 때마다 자동으로 증가합니다. OS 이미지의 경우 저장소에 여러 리비전이 함께 있을 수 있습니다.

컨테이너 이미지의 경우 저장소에는 최신 리비전만 보존됩니다. 이전 버전(패키지, 패치 등)에 대한 정보는 보존되며 **구식 표시** 확인란을 사용하여 나열할 수 있습니다.

이미지에서 **[자세히]** 버튼을 클릭하면 자세한 내용을 확인할 수 있습니다. 상세 보기에는 관련 패치의 정확한 목록, 이미지 내에 설치된 모든 패키지 목록 및 빌드 로그가 포함됩니다.

[삭제] 버튼을 클릭하면 목록에서 이미지가 삭제됩니다. 또한, 관련 열, OS 이미지 저장소의 파일 및 사용되지 않는 리비전도 삭제됩니다.

빌드가 성공한 후 상태를 검사한 경우에만 패치 및 패키지 목록을 사용할 수 있습니다.

Chapter 12. 인프라 유지보수 작업

예정된 작동 중지 시간 기간 관련 작업 시 SUSE Multi-Linux Manager 서버의 중요한 작동 중지 시간 전, 도중, 이후에 수행해야 하는 모든 작업을 기억하는 것은 어려울 수 있습니다. 서버 간 동기화 슬레이브 서버 또는 SUSE Multi-Linux Manager 프록시 등 SUSE Multi-Linux Manager 서버 관련 시스템도 영향을 받으므로 고려해야 합니다.

SUSE는 항상 SUSE Multi-Linux Manager 인프라를 최신 상태로 유지할 것을 권장합니다. 여기에는 서버, 프록시 및 빌드 호스트가 포함됩니다. SUSE Multi-Linux Manager 서버를 업데이트된 상태로 유지하지 않으면 필요할 때 환경의 일부를 업데이트하지 못할 수 있습니다.

이 섹션에는 각 단계를 수행하는 추가 정보 링크와 작동 중지 시간 기간에 실행하는 검사 목록이 포함되어 있습니다.

12.1. 서버

절차: 서버 확인

1. 최신 업데이트를 적용합니다.
2. 필요한 경우, 최신 서비스 팩으로 업그레이드합니다.
3. Run **podman ps** and check whether all required services are up and running.

On SUSE Linux Enterprise Server 15 SP7, you can install updates using a package manager:

- YaST 사용에 대한 자세한 내용은 https://documentation.suse.com/sles/15-SP7/html/SLES-all/cha-onlineupdate-you.html에서 확인할 수 있습니다.
- zypper 사용에 대한 자세한 내용은 https://documentation.suse.com/sles/15-SP7/html/SLES-all/cha-sw-cl.html#sec-zypper에서 확인할 수 있습니다.

On SL Micro 6.1, you can install updates using the **transactional-update** command. For information on using **transactional-update**, see https://documentation.suse.com/sle-micro/6.1/html/Micro-transactional-updates/transactional-updates.html.

기본적으로 SUSE Multi-Linux Manager 서버에 대해 여러 업데이트 채널이 구성 및 활성화되어 있습니다. 새 패키지와 업데이트된 패키지를 자동으로 사용할 수 있습니다.

SUSE Multi-Linux Manager을 최신 상태로 유지하려면 SUSE Customer Center에 직접 연결하거나 Repository Management Tool(RMT)을(를) 사용합니다. RMT를 연결이 끊긴 환경의 로컬 설치 소스로 사용할 수 있습니다.

시스템에서 업데이트 채널을 사용할 수 있는지 확인할 수 있는 명령은 다음과 같습니다.

```
zypper lr
```

다음과 유사한 형태로 출력됩니다.

```
Name                                                | Enabled | GPG Check | Refresh
```

```
-----------------------------------------------------+---------+-----------+--------
SLE-Module-Basesystem15-SP7-Pool                     | Yes     | (r ) Yes  | No
SLE-Module-Basesystem15-SP7-Updates                  | Yes     | (r ) Yes  | Yes
SLE-Module-Containers15-SP7-Pool                     | Yes     | (r ) Yes  | No
SLE-Module-Containers15-SP7-Updates                  | Yes     | (r ) Yes  | Yes
SLE-Module-Python3-15-SP7-Pool                       | Yes     | (r ) Yes  | No
SLE-Module-Python3-15-SP7-Updates                    | Yes     | (r ) Yes  | Yes
SLE-Product-SLES15-SP7-Pool                          | Yes     | (r ) Yes  | No
SLE-Product-SLES15-SP7-Updates                       | Yes     | (r ) Yes  | Yes
SUSE-Multi-Linux-Manager-Server-SLE-5.1-Pool         | Yes     | (r ) Yes  | No
SUSE-Multi-Linux-Manager-Server-SLE-5.1-Updates      | Yes     | (r ) Yes  | Yes
SLE-Module-Server-Applications15-SP7-Pool            | Yes     | (r ) Yes  | No
SLE-Module-Server-Applications15-SP7-Updates         | Yes     | (r ) Yes  | Yes
SLE-Module-Systems-Management-15-SP7-Pool            | Yes     | (r ) Yes  | No
SLE-Module-Systems-Management-15-SP7-Updates         | Yes     | (r ) Yes  | Yes
```

SUSE Multi-Linux Manager releases maintenance updates (MUs) to provide newer packages. Maintenance updates are indicated with a new version number. For example, the major release 5.1 is incremented to 5.1.1 when an MU is released.

Web UI의 탐색 모음 하단을 보면 실행 중인 버전을 확인할 수 있습니다. **api.getVersion()** XMLRPC API 호출로 버전 번호를 가져올 수도 있습니다.

12.1.1. 클라이언트 도구

서버가 업데이트되면 클라이언트의 일부 도구도 업데이트하는 것이 좋습니다. 클라이언트에서 **salt-minion**, **zypper** 및 기타 관련 관리 패키지를 업데이트하는 것은 엄격한 요구 사항이 아니지만 일반적으로 모범 사례입니다. 예를 들어, 서버의 유지보수 업데이트에서는 주요 새 Salt 버전이 도입될 수 있습니다. 그러한 경우 Salt 클라이언트는 계속 작동하지만 나중에 문제가 발생할 수 있습니다. 이를 방지하기 위해 SUSE는 **venv-salt-minion**이 항상 안전하게 업데이트될 수 있도록 보장합니다.

12.2. 서버 간 동기화 슬레이브 서버

서버 간 동기화 슬레이브 서버를 사용 중인 경우 SUSE Multi-Linux Manager 서버 업데이트 완료 후 업데이트합니다.

자세한 내용은 **Specialized-guides › Large-deployments**에서 확인할 수 있습니다.

12.3. 모니터링 서버

Prometheus용 모니터링 서버를 사용 중인 경우 SUSE Multi-Linux Manager 서버 업데이트가 완료된 후 업데이트하십시오.

모니터링에 대한 자세한 내용은 **Administration › Monitoring**에서 확인할 수 있습니다.

12.4. 프록시

SUSE Multi-Linux Manager 서버 업데이트가 완료되는 즉시 프록시를 업데이트해야 합니다.

일반적으로 다른 버전의 서버에 연결된 프록시를 실행하는 기능은 지원되지 않습니다. 유일한 예외는 서버가 먼저 업데이트되어 프록시가 이전 버전을 일시적으로 실행할 수 있을 것으로 예상되는 업데이트 기간 동안입니다.

항상 서버를 먼저 업그레이드한 다음 프록시를 업그레이드하십시오.

Chapter 13. SUSE Multi-Linux Manager을(를) 사용한 라이브 패치

커널 업데이트를 수행하려면 일반적으로 시스템을 재부팅해야 합니다. CVE(Common vulnerability and exposure) 패치는 최대한 빨리 적용해야 하지만, 작동 중지 시간을 감당할 수 없는 경우 라이브 패치를 사용하여 이러한 중요한 업데이트를 삽입하고 재부팅하지 않아도 됩니다.

SLES 12와 SLES 15 간에는 라이브 패치 설정 절차가 약간 다릅니다. 두 절차 모두 이 섹션에 설명되어 있습니다.

13.1. 라이브 패치를 위한 채널 설정

전체 커널 패키지를 업데이트할 때마다 재부팅해야 합니다. 그러므로 라이브 패치를 사용하는 클라이언트의 경우 할당된 채널에서 사용할 수 있는 최신 커널이 없는 것이 중요합니다. 라이브 패치를 사용하는 클라이언트에는 라이브 패치 채널에서 실행 중인 커널에 대한 업데이트가 있습니다.

라이브 패치를 위한 채널을 관리하는 두 가지 방법이 있습니다.

컨텐트 라이프싸이클 관리를 사용하여 제품 트리를 복제하고 실행 중인 버전보다 최신인 커널 버전을 제거합니다. 이 절차에 대한 설명은 administration:content-lifecycle-examples.pdf에서 확인할 수 있습니다. 이것이 권장되는 솔루션입니다.

또는 **spacewalk-manage-channel-lifecycle** 도구를 사용합니다. 절차는 수동으로 진행되며 명령 줄 도구 및 Web UI를 필요로 합니다. 이 섹션에서는 SLES 15 SP5에서 작동하는 절차를 설명하지만 SLE 12 SP4 이상에서도 작동합니다.

13.1.1. 라이브 패치에 spacewalk-manage-channel-lifecycle 사용

spacewalk-manage-channel-lifecycle은 더 이상 사용되지 않으며 다음 릴리스에서 제거될 예정입니다. 대신 풍부한 기능을 지원하는 컨텐트 라이프사이클 관리(CLM) API로 전환하는 것이 좋습니다.

복제된 벤더 채널은 개발의 경우 **dev**, 프로덕션의 경우 **testing** 또는 **prod**로 접두사를 지정해야 합니다. 이 절차에서는 **dev** 복제된 채널을 생성한 후, 채널을 **testing**으로 승격하겠습니다.

절차: 라이브 패치 채널 복제

1. 클라이언트의 명령 프롬프트에서 루트 권한으로 현재 패키지 채널 트리를 가져옵니다.

```
# spacewalk-manage-channel-lifecycle --list-channels
Spacewalk 사용자 이름: admin
Spacewalk 비밀번호:
채널 트리:

 1. sles15-sp7-pool-x86_64
      \__ sle-live-patching15-pool-x86_64-sp7
      \__ sle-live-patching15-updates-x86_64-sp7
      \__ sle-manager-tools15-pool-x86_64-sp7
      \__ sle-manager-tools15-updates-x86_64-sp7
      \__ sles15-sp7-updates-x86_64
```

2. **init** 인수와 함께 **spacewalk-manage-channel** 명령을 사용하여 기존 벤더 채널의 새 개발 클론을 자동으로 생성합니다.

```
spacewalk-manage-channel-lifecycle --init -c sles15-sp7-pool-x86_64
```

3. 채널 목록에서 **dev-sles15-sp7-updates-x86_64**를 사용할 수 있는지 확인합니다.

생성한 **dev** 복제 채널을 확인하고 재부팅이 필요한 커널 업데이트를 제거합니다.

절차: 복제된 채널에서 비라이브 커널 패치 제거하기

1. **시스템 › 시스템 목록**에서 클라이언트를 선택하고 **커널** 필드에 표시된 버전을 기록하여 현재 커널 버전을 확인합니다.
2. SUSE Multi-Linux Manager Web UI의 **시스템 › 개요**에서 클라이언트를 선택하고 **소프트웨어 › 관리 › 채널** 탭으로 이동한 후 **dev-sles15-sp7-updates-x86_64**를 선택합니다. **패치** 탭으로 이동하여 **[패치 나열/제거]**를 클릭합니다.
3. 검색 창에 **kernel**을 입력하고 현재 클라이언트에서 사용하는 커널과 일치하는 커널 버전을 찾습니다.
4. 현재 설치된 커널 버전보다 최신 커널 버전을 모두 제거합니다.

이제 채널이 라이브 패치용으로 설정되었으니 **testing**으로 승격할 수 있습니다. 이 절차에서는 적용할 준비가 된 라이브 패치 하위 채널도 클라이언트에 추가합니다.

절차: 라이브 패치 채널 승격

1. 클라이언트의 명령 프롬프트에서 **루트** 권한으로 **dev-sles15-sp7-pool-x86_64** 채널을 새 **testing** 채널로 승격 및 복제합니다.

```
# spacewalk-manage-channel-lifecycle --promote -c dev-sles15-sp7-pool-x86_64
```

2. SUSE Multi-Linux Manager Web UI의 **시스템 › 개요**에서 클라이언트를 선택하고 **소프트웨어 › 소프트웨어 채널** 탭으로 이동합니다.
3. 새 **test-sles15-sp7-pool-x86_64** 사용자 정의 채널을 확인하여 기본 채널을 변경하고 해당하는 라이브 패치 하위 채널을 모두 확인합니다.
4. **[다음]**을 클릭하고 세부 사항이 올바른지 확인한 후 **[확인]**을 클릭하여 변경 사항을 저장합니다.

이제 사용할 수 있는 CVE 패치를 선택하여 볼 수 있으며, 라이브 패치를 사용하여 이러한 중요한 커널 업데이트를 적용할 수 있습니다.

13.2. SLES 15의 라이브 패치

SLES 15 이상 시스템에서 라이브 패치는 **klp livepatch** 도구로 관리됩니다.

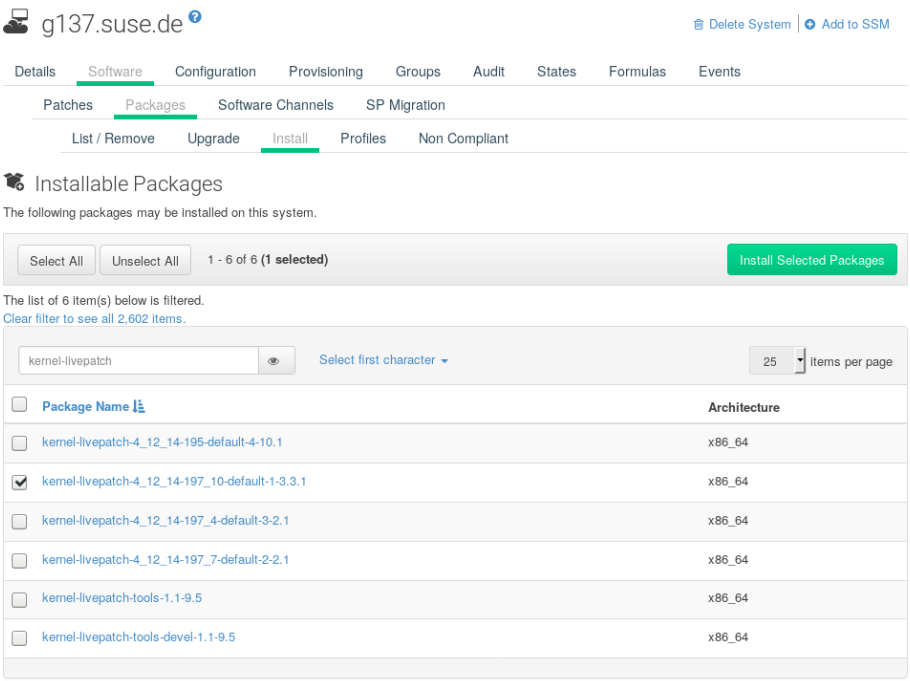
시작하기 전, 다음을 확인합니다.

- SUSE Multi-Linux Manager가 완전히 업데이트되었습니다.

- SLES 15(SP1 이상)를 실행하는 Salt 클라이언트가 한 개 이상 있습니다.
- SLES 15 Salt 클라이언트가 SUSE Multi-Linux Manager에 등록되었습니다.
- 라이브 패치 하위 채널을 포함하여 아키텍처에 적합한 SLES 15 채널에 액세스할 수 있습니다.
- 클라이언트가 완전히 동기화되었습니다.
- 라이브 패치를 위해 준비된 복제된 채널에 클라이언트를 할당합니다. 준비에 대한 자세한 내용은 **Administration › Live-patching-channel-setup**에서 확인할 수 있습니다.

절차: 라이브 패치 설정

1. **시스템 › 개요**에서 라이브 패치를 사용하여 관리할 클라이언트를 선택하고 **소프트웨어 › 패키지 › 설치** 탭으로 이동합니다. **kernel-livepatch** 패키지를 검색하여 설치합니다.



2. 라이브 패치를 활성화하려면 highstate를 적용하고 클라이언트를 재부팅합니다.
3. 라이브 패치로 관리할 각 클라이언트에 대해 반복합니다.
4. 라이브 패치가 올바르게 활성화되었는지 확인하려면 **시스템 › 시스템 목록**에서 클라이언트를 선택하고 **커널** 필드에 **라이브 패치**가 표시되는지 확인합니다.

절차: 커널에 라이브 패치 적용

1. SUSE Multi-Linux Manager Web UI의 **시스템 › 개요**에서 클라이언트를 선택합니다. 화면 상단의 배너는 클라이언트에서 사용할 수 있는 중요 패키지와 중요하지 않은 패키지의 수를 표시합니다.
2. 사용할 수 있는 중요 패치 목록을 살펴보려면 **[중요]**를 클릭합니다.
3. **중요: Linux 커널용 보안 업데이트**라는 설명이 있는 패치를 선택합니다. 보안 버그에는 CVE 번호도 포함될 수 있습니다.
4. 선택 사항: 적용할 패치의 CVE 번호를 알고 있는 경우 **감사 › CVE 감사**에서 검색하여 패치가 필요한 모든 클라이언트에 패치를 적용할 수 있습니다.

- 모든 커널 패치가 라이브 패치는 아닙니다. 비라이브 커널 패치는 **보안** 방패 아이콘 옆에 **재부팅 필요** 아이콘이 표시됩니다. 이러한 패치는 항상 재부팅해야 합니다.
- 라이브 패치를 적용하여 모든 보안 문제를 해결할 수 있는 것은 아닙니다. 일부 보안 문제는 전체 커널 업데이트를 적용해야만 수정할 수 있으며 재부팅해야 합니다. 이러한 문제에 대해 할당된 CVE 번호는 라이브 패치에 포함되지 않습니다. CVE 감사에는 이러한 요구 사항이 표시됩니다.

13.3. SLES 12의 라이브 패치

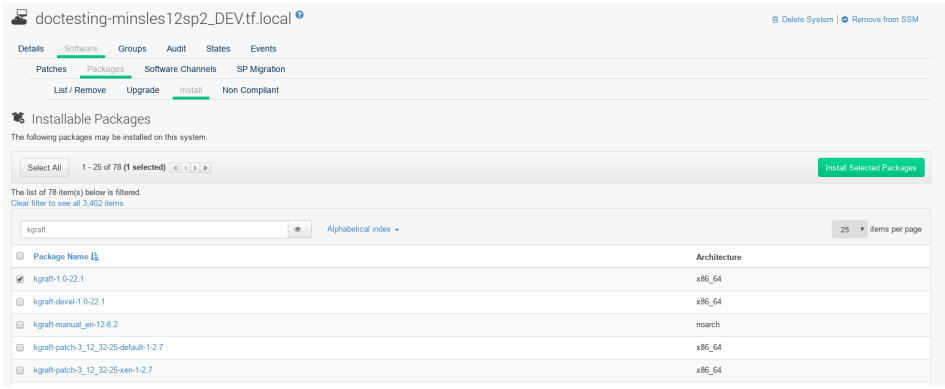
SLES 12 시스템에서 라이브 패치는 kGraft에서 관리합니다. kGraft 사용에 대한 자세한 내용은 https://documentation.suse.com/sles/12-SP5/html/SLES-all/cha-kgraft.html을 참조하십시오.

시작하기 전, 다음을 확인합니다.

- SUSE Multi-Linux Manager가 완전히 업데이트되었습니다.
- SLES 12(SP1 이상)를 실행하는 Salt 클라이언트가 한 개 이상 있습니다.
- SLES 12 Salt 클라이언트가 SUSE Multi-Linux Manager에 등록되었습니다.
- 라이브 패치 하위 채널 등 아키텍처에 적합한 SLES 12 채널에 액세스할 수 있습니다.
- 클라이언트가 완전히 동기화되었습니다.
- 라이브 패치를 위해 준비된 복제된 채널에 클라이언트를 할당합니다. 준비에 대한 자세한 내용은 **Administration › Live-patching-channel-setup**에서 확인할 수 있습니다.

절차: 라이브 패치 설정

1. **시스템 › 개요**에서 라이브 패치를 사용하여 관리할 클라이언트를 선택하고, 시스템 세부 사항 페이지에서 **소프트웨어 › 패키지 › 설치** 탭으로 이동합니다. **kgraft** 패키지를 검색하여 설치합니다.



2. 라이브 패치를 활성화하려면 highstate를 적용하고 클라이언트를 재부팅합니다.
3. 라이브 패치로 관리할 각 클라이언트에 대해 반복합니다.
4. 라이브 패치가 올바르게 활성화되었는지 확인하려면, **시스템 › 시스템 목록**에서 클라이언트를 선택하고 **라이브 패치**가 **커널** 필드에 표시되는지 확인합니다.

절차: 커널에 라이브 패치 적용

1. SUSE Multi-Linux Manager Web UI의 **시스템 › 개요**에서 클라이언트를 선택합니다. 화면 상단의 배너는 클라이언트에서 사용할 수 있는 중요 패키지와 중요하지 않은 패키지의 수를 표시합니다.
2. 사용할 수 있는 중요 패치 목록을 살펴보려면 **[중요]**를 클릭합니다.
3. **중요: Linux 커널용 보안 업데이트**라는 설명이 있는 패치를 선택합니다. 보안 버그에는 CVE 번호도 포함될 수 있습니다.
4. 선택 사항: 적용할 패치의 CVE 번호를 알고 있는 경우 **감사 › CVE 감사**에서 검색하여 패치가 필요한 모든 클라이언트에 패치를 적용할 수 있습니다.

- 모든 커널 패치가 라이브 패치는 아닙니다. 비라이브 커널 패치는 **보안** 방패 아이콘 옆에 **재부팅 필요** 아이콘이 표시됩니다. 이러한 패치는 항상 재부팅해야 합니다.
- 라이브 패치를 적용하여 모든 보안 문제를 해결할 수 있는 것은 아닙니다. 일부 보안 문제는 전체 커널 업데이트를 적용해야만 수정할 수 있으며 재부팅해야 합니다. 이러한 문제에 대해 할당된 CVE 번호는 라이브 패치에 포함되지 않습니다. CVE 감사에는 이러한 요구 사항이 표시됩니다.

Chapter 14. 유지보수 기간

SUSE Multi-Linux Manager의 유지보수 기간 기능을 사용하면 예약된 유지보수 기간 동안 수행할 작업을 예약할 수 있습니다. 유지보수 기간 일정을 생성하여 클라이언트에 적용하면 지정된 기간 외에 일부 작업을 실행할 수 없습니다.

유지보수 기간의 작동 방식은 시스템 잠금의 방식과 다릅니다. 시스템 잠금은 필요에 따라 켜거나 끌 수 있지만, 유지보수 기간은 작업이 허용되는 기간을 정의합니다. 또한, 허용되는 작업과 제한된 작업이 다릅니다. 시스템 잠금에 대한 자세한 내용은 **Client-configuration › System-locking**에서 확인할 수 있습니다.

유지보수 기간에는 달력 및 일정이 모두 필요합니다. 달력은 정기적 이벤트 등 유지보수 기간 이벤트의 날짜와 시간을 정의하며 **ical** 형식이어야 합니다. 일정은 달력에 정의된 이벤트를 사용하여 유지보수 기간을 생성합니다. **ical** 파일을 생성하여 업로드하거나 기존 **ical** 파일에 링크하여 달력을 생성한 후에만 일정을 생성할 수 있습니다.

일정을 생성하면 SUSE Multi-Linux Manager 서버에 등록된 클라이언트에 할당할 수 있습니다. 유지보수 일정이 할당된 클라이언트는 유지보수 기간 외에 제한된 작업을 실행할 수 없습니다.

제한되는 작업은 클라이언트를 대폭 수정하고 클라이언트 실행을 중지할 가능성이 있습니다. 제한되는 작업의 몇 가지 예는 다음과 같습니다.

- 패키지 설치
- 클라이언트 업그레이드
- 제품 마이그레이션
- Highstate 애플리케이션

제한되지 않는 작업은 안전한 작업으로 간주되고 클라이언트에 문제를 유발한 가능성이 낮은 소규모 작업입니다. 제한되지 않는 작업의 몇 가지 예는 다음과 같습니다.

- 패키지 프로파일 업데이트
- 하드웨어 새로 고침
- 소프트웨어 채널 구독

시작하기 전, 업로드할 **ical** 파일을 생성하거나 달력을 생성하려면 **ical** 파일에 링크해야 합니다. Microsoft Outlook, Google Calendar 또는 KOrganizer와 같은 기본 달력 도구에서 **ical** 파일을 생성할 수 있습니다.

절차: 새 유지보수 달력 업로드

1. SUSE Multi-Linux Manager Web UI에서 **일정 › 유지보수 기간 › 달력**으로 이동하고 **[생성]**을 클릭합니다.
2. **달력 이름** 섹션에 달력 이름을 입력합니다.
3. **ical** 파일에 대한 URL을 입력하거나 파일을 직접 업로드합니다.
4. **[달력 생성]**을 클릭하여 달력을 저장합니다.

절차: 새 일정 생성

1. SUSE Multi-Linux Manager Web UI에서 **일정 › 유지보수 기간 › 일정**으로 이동하고 **[생성]**을 클릭합니다.

2. **일정 이름** 섹션에 일정 이름을 입력합니다.
3. 선택 사항: **ical** 파일에 두 개 이상의 일정에 적용되는 이벤트가 포함된 경우 **다중**을 선택합니다.
4. 이 일정에 할당할 달력을 선택합니다.
5. **[일정 생성]**을 클릭하여 일정을 저장합니다.

절차: 클라이언트에 일정 할당

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록**으로 이동하여 일정에 할당할 클라이언트를 선택하고 **시스템 속성** 패널을 찾은 후 **[이러한 속성 편집]**을 클릭합니다. 또는, **시스템 › 시스템 세트 관리자**로 이동하고 **기타 › 유지보수 기간** 탭을 사용하여 시스템 세트 관리자를 통해 클라이언트를 할당할 수 있습니다.
2. **시스템 세부 사항 편집** 페이지에서 **유지보수 일정** 필드를 찾아 할당할 일정의 이름을 선택합니다.
3. **[속성 업데이트]**를 클릭하여 유지보수 일정을 할당합니다.

클라이언트에 새 유지보수 일정을 할당할 때 클라이언트에 이미 일부 제한된 작업이 예약되어 있을 수 있으며, 그러한 경우 이러한 작업이 새 유지보수 일정과 충돌할 수 있습니다. 이러한 경우 Web UI에 오류가 표시되고 클라이언트에 일정을 할당할 수 없습니다. 이를 해결하려면 일정을 할당할 때 **[영향을 받는 작업 취소]** 옵션을 선택합니다. 그러면 새 유지보수 일정과 충돌하는 모든 이전 예약 작업이 취소됩니다.

유지보수 기간을 생성하면 패키지 업그레이드와 같은 제한된 작업이 유지보수 기간 동안 수행되도록 예약할 수 있습니다.

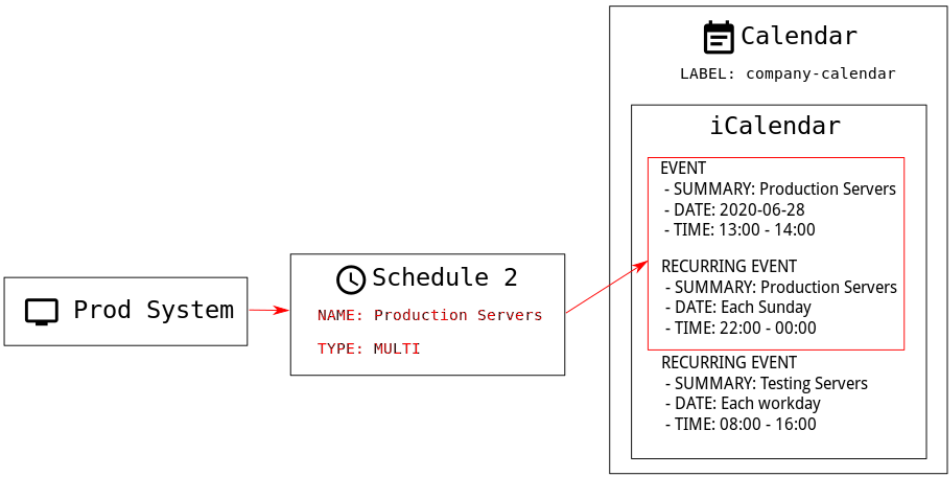
절차: 패키지 업그레이드 예약

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록**으로 이동하여 업그레이드할 클라이언트를 선택한 후 **소프트웨어 › 패키지 › 업그레이드** 탭으로 이동합니다.
2. 목록에서 업그레이드할 패키지를 선택하고 **[패키지 업그레이드]**를 클릭합니다.
3. **유지보수 기간** 필드에서 클라이언트가 업그레이드를 수행하기 위해 사용해야 하는 유지보수 기간을 선택합니다.
4. 패키지 업그레이드를 예약하려면 **[확인]**을 클릭합니다.

14.1. 유지보수 일정 유형

달력을 생성할 때 일회성 이벤트 또는 정기적 이벤트일 수 있는 여러 이벤트가 포함됩니다. 각 이벤트에는 **요약** 필드가 있습니다. 달력 하나에 유지보수 일정을 여러 개 생성하려면 **요약** 필드를 사용하여 이벤트를 각각 지정하면 됩니다.

예를 들어, 프로덕션 서버에 대한 일정을 생성하고 서버 테스트에 대해 다른 일정을 생성할 수 있습니다. 이 경우 프로덕션 서버의 이벤트에는 **요약: 프로덕션 서버**를 지정하고 테스트 서버의 이벤트에는 **요약: 테스트 서버**를 지정합니다.



일정에는 단일 또는 다중의 두 가지 유형이 있습니다. 두 개 이상의 일정에 적용되는 이벤트가 달력에 포함된 경우 **다중**을 선택하고 달력 파일에서 사용한 **요약** 필드에 따라 일정 이름을 지정해야 합니다.

절차: 다중 일정 생성

1. SUSE Multi-Linux Manager Web UI에서 **일정 › 유지보수 기간 › 일정**으로 이동하고 **[생성]**을 클릭합니다.
2. **일정 이름** 섹션에서 일정 이름을 입력합니다. 달력의 **요약** 필드와 일치하는지 확인합니다.
3. **다중** 옵션을 확인합니다.
4. 이 일정에 할당할 달력을 선택합니다.
5. **[일정 생성]**을 클릭하여 일정을 저장합니다.
6. 다음 일정을 생성하려면 **[생성]**을 클릭합니다.
7. **일정 이름** 섹션에 두 번째 일정의 이름을 입력합니다. 두 번째 달력의 **요약** 필드와 일치하는지 확인합니다.
8. **다중** 옵션을 확인합니다.
9. **[일정 생성]**을 클릭하여 일정을 저장합니다.
10. 생성해야 할 각 일정을 반복합니다.

14.2. 제한되는 작업 및 제한되지 않는 작업

이 섹션에는 제한되는 작업 및 제한되지 작업의 전체 목록이 포함되어 있습니다.

제한되는 작업은 클라이언트를 대폭 수정하고 클라이언트 실행을 중지할 가능성이 있습니다. 제한되는 작업은 유지보수 기간에만 실행할 수 있습니다. 제한되는 작업은 다음과 같습니다.

- 패키지 작업(예: 패키지 설치, 업데이트 또는 제거)
- 패치 업데이트
- 클라이언트 재부팅
- 트랜잭션 롤백
- 구성 관리 변경 작업

- highstate 적용
- 자동 설치 및 재설치
- 원격 명령
- 제품 마이그레이션
- 클러스터 작업

Salt › 원격 명령으로 이동하여 언제든지 직접 원격 명령을 실행할 수 있습니다. 이는 클라이언트의 유지보수 기간 여부와 관계없이 적용됩니다. 원격 명령에 대한 자세한 내용은 **Administration › Actions**에서 확인할 수 있습니다.

Unrestricted actions are minor actions that are considered safe and are unlikely to cause problems on the client. If an action is not restricted it is, by definition, unrestricted, and can be run at any time.

Chapter 15. mgr-sync 사용

mgr-sync 도구는 명령 프롬프트에서 사용되며, Web UI에서 일부 경우 사용할 수 없는 SUSE Multi-Linux Manager를 사용하기 위한 기능을 제공합니다. **mgr-sync**의 주요 용도는 SUSE Customer Center에 연결하고, 제품 및 패키지 정보를 검색하며, SUSE Multi-Linux Manager 서버와의 동기화를 위해 채널을 준비하는 것입니다.

이 도구는 SUSE 지원 구독과 함께 사용하도록 설계되었으며, openSUSE, CentOS 및 Ubuntu 등 오픈 소스 배포에는 필요하지 않습니다.

mgr-sync에서 사용할 수 있는 명령과 인수는 이 테이블에서 확인할 수 있습니다. **mgr-sync** 명령에 다음 구문을 사용합니다.

```
mgr-sync [-h] [--version] [-v] [-s] [-d {1,2,3}] {list,add,refresh,delete}
```

표 3. mgr-sync 명령

명령	설명	예시 사용
list	채널, 조직 자격 증명 또는 제품 나열	**mgr-sync list channels**
add	채널, 조직 자격 증명 또는 제품 추가	**mgr-sync add channel <channel_name>**
refresh	제품, 채널 및 구독의 로컬 사본 새로 고침	**mgr-sync refresh**
delete	로컬 시스템에서 기존 SCC 조직 자격 증명 삭제	**mgr-sync delete credentials**
sync	지정된 채널 동기화 또는 공백으로 두면 요청	**mgr-sync sync channel <channel_name>**

명령과 관련된 전체 옵션 목록을 확인하려면 다음 명령을 사용합니다.

```
mgr-sync <command> --help
```

표 4. mgr-sync 선택 사항 인수

옵션	약식 옵션	설명	예시 사용
help	**h**	명령 사용법 및 옵션 표시	**mgr-sync --help**
version	N/A	현재 설치된 **mgr-sync** 버전 표시	**mgr-sync --version**
verbose	**v**	자세한 출력 제공	**mgr-sync --verbose refresh**
store-credentials	**s**	자격 증명을 로컬 숨김 파일에 저장	**mgr-sync --store -credentials**

옵션	약식 옵션	설명	예시 사용
debug	**d**	추가 디버깅 정보 기록. 1, 2, 3의 수준 필요. 3은 가장 많은 양의 디버깅 정보를 제공	**mgr-sync -d 3 refresh**
no-sync	N/A	동기화를 시작하지 않고 제품 또는 채널을 추가하려면 **add** 명령과 함께 사용	**mgr-sync --no-sync add <channel_name>**

다음은 **mgr-sync**에 대한 로그의 위치입니다.

- **/var/lib/containers/storage/volumes/var-log/_data/rhn/mgr-sync.log**
- **/var/lib/containers/storage/volumes/var-log/_data/rhn/rhn_web_api.log**

Chapter 16. Prometheus 및 Grafana를 사용한 모니터링

Prometheus 및 Grafana를 사용하여 SUSE Multi-Linux Manager 환경을 모니터링할 수 있습니다. SUSE Multi-Linux Manager 서버 및 프록시는 자체 상태 메트릭을 제공할 수 있습니다. 또한, Salt 클라이언트에서 여러 Prometheus 엑스포트를 설치 및 관리할 수 있습니다.

16.1. 요구사항

Prometheus 및 Grafana 패키지가 SUSE Multi-Linux Manager 클라이언트 도구에 포함된 대상은 다음과 같습니다.

- SUSE Linux Enterprise 12
- SUSE Linux Enterprise 15
- openSUSE Leap 15.x

위에 나열된 클라이언트만 모니터링 서버로 지원됩니다.

SUSE Multi-Linux Manager 서버와 별도의 시스템에 Prometheus 및 Grafana를 설치해야 합니다. 관리형 Salt SUSE 클라이언트를 모니터링 서버로 사용하는 것이 좋습니다.

Prometheus는 풀 방식을 사용하여 메트릭을 가져오므로 서버는 모니터링되는 클라이언트에 대한 TCP 연결을 설정할 수 있어야 합니다. 클라이언트에는 해당하는 열린 포트가 있어야 하며 네트워크를 통해 연결할 수 있어야 합니다. 또는, 역방향 프록시를 사용하여 연결을 설정할 수 있습니다.

모니터링할 각 클라이언트에 대한 모니터링 추가 기능 구독이 있어야 합니다. SUSE Multi-Linux Manager 구독을 관리하려면 SUSE Customer Center을(를) 방문하십시오.

16.2. Prometheus 및 Grafana

16.2.1. Prometheus

Prometheus는 시계열 데이터베이스에 실시간 메트릭을 기록하기 위해 사용되는 오픈 소스 모니터링 도구입니다. 메트릭은 HTTP를 통해 끌어오며, 고성능과 확장성을 가능하게 합니다.

Prometheus 메트릭은 시계열 데이터 또는 동일한 그룹 또는 차원에 속하는 타임스탬프 값입니다. 메트릭은 이름과 레이블 집합으로 고유하게 식별됩니다.

```
  메트릭 이름                 레이블                    타임스탬프      값
┌──────────────┐┌──────────────────────┐   ┌──────────────────────────┐┌──┐
┌──────────────┐└──────────────────────┘   └──────────────────────────┘└──┘
http_requests_total{status="200", method="GET"}  @1557331801.111  42236
```

모니터링되는 각 애플리케이션 또는 시스템은 코드 계측 또는 Prometheus 엑스포트를 통해 위의 형식으로 메트릭을 노출해야 합니다.

16.2.2. Prometheus 엑스포트

엑스포트는 타사 시스템의 메트릭을 Prometheus 메트릭으로 엑스포트할 때 유용하게 사용할 수 있는 라이브러리입니다. 엑스포트는 Prometheus 메트릭을 사용하여 특정 애플리케이션 또는 시스템을 직접 계측하는 것이 불가능할 때마다 유용합니다. 모니터링되는 호스트에서 여러 엑스포트를 실행하여 로컬 메트릭을 엑스포트할 수 있습니다.

Prometheus 커뮤니티는 공식 엑스포트 목록을 제공하며, 커뮤니티 기여를 통해 더욱 추가될 것입니다. 자세한 내용과 광범위한 엑스포트 목록은 https://prometheus.io/docs/instrumenting/exporters/에서 확인할 수 있습니다.

16.2.3. Grafana

Grafana는 데이터 시각화, 모니터링 및 분석을 위한 도구입니다. 그리고 일정 기간 동안 특정 메트릭을 나타내는 패널이 포함된 대시보드를 생성하기 위한 용도로 사용됩니다. 일반적으로 Grafana는 Prometheus와 함께 사용되지만, ElasticSearch, MySQL, PostgreSQL 및 Influx DB와 같은 다른 데이터 소스도 지원합니다. Grafana에 대한 자세한 내용은 https://grafana.com/docs/에서 확인할 수 있습니다.

16.3. 모니터링 서버 설정

모니터링 서버를 설정하려면 Prometheus와 Grafana를 설치 및 구성해야 합니다.

SUSE 클라이언트만 모니터링 서버로 지원됩니다. 전체 목록은 administration:monitoring.pdf에서 확인할 수 있습니다.

16.3.1. Prometheus 설치

모니터링 서버가 Salt 클라이언트인 경우 SUSE Multi-Linux Manager Web UI를 사용하여 Prometheus 패키지를 설치할 수 있습니다. 그렇지 않으면 모니터링 서버에 패키지를 수동으로 다운로드하여 설치할 수 있습니다. Prometheus 소프트웨어는 SUSE Multi-Linux Manager 프록시 및 SUSE Multi-Linux Manager for Retail 분기 서버에서도 사용할 수 있습니다.

- SUSE Multi-Linux Manager 서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행하거나, 단일 명령을 실행하려면 **mgrctl exec <options> — <command>**를 실행하십시오.
- 컨테이너 내부에서 컨테이너 호스트로 파일을 복사하려면 **mgrctl cp**를 사용합니다.

데이터를 저장하려면 Prometheus에 POSIX 파일 시스템이 필요합니다. POSIX 호환 파일 시스템이 아닌 파일 시스템은 지원되지 않습니다. NFS 파일 시스템은 지원되지 않습니다.

절차: Web UI를 사용한 Prometheus 설치

1. SUSE Multi-Linux Manager Web UI에서 Prometheus를 설치할 시스템의 세부 사항 페이지를 열고 **수식** 탭으로 이동합니다.
2. **Prometheus** 확인란을 선택하여 모니터링 수식을 활성화하고 [**저장**]을 클릭합니다.
3. 상단 메뉴의 **Prometheus** 탭으로 이동합니다.

4. **SUSE Multi-Linux Manager 서버** 섹션에 유효한 SUSE Multi-Linux Manager API 자격 증명을 입력합니다. 입력한 자격 증명이 모니터링 대상 시스템 집합에 대한 액세스를 허용하는지 확인합니다.
5. 필요에 따라 기타 구성 옵션을 사용자 정의합니다.
6. **[수식 저장]**을 클릭합니다.
7. highstate를 적용하고 성공적으로 완료되었는지 확인합니다.
8. Prometheus 인터페이스를 올바르게 로드했는지 확인합니다. 브라우저에서 9090 포트 Prometheus가 설치된 서버의 URL(예: http://example.com:9090)로 이동합니다.

수식 모니터링에 대한 자세한 내용은 **Specialized-guides › Salt**에서 참조하십시오.

절차: 수동으로 Prometheus 설치 및 구성

1. 모니터링 서버에서 **golang-github-prometheus-prometheus** 패키지 설치:

```
zypper in golang-github-prometheus-prometheus
```

2. Prometheus 서비스를 활성화합니다.

```
systemctl enable --now prometheus
```

3. Prometheus 인터페이스를 올바르게 로드했는지 확인합니다. 브라우저에서 9090 포트 Prometheus가 설치된 서버의 URL(예: http://example.com:9090)로 이동합니다.
4. **/etc/prometheus/prometheus.yml**의 구성 파일을 열고 이 구성 정보를 추가하십시오. **server.url**을 SUSE Multi-Linux Manager 서버 URL로 바꾸고 SUSE Multi-Linux Manager 자격 증명과 일치하도록 **사용자 이름** 및 **비밀번호** 필드를 조정합니다.

```
# {productname} self-health metrics
scrape_configs:
- job_name: 'mgr-server'
  static_configs:
    - targets:
      - 'server.url:9100'  # Node exporter
      - 'server.url:9187'  # PostgreSQL exporter
      - 'server.url:5556'  # JMX exporter (Tomcat)
      - 'server.url:5557'  # JMX exporter (Taskomatic)
      - 'server.url:9800'  # Taskomatic
    - targets:
      - 'server.url:80'    # Message queue
      labels:
        __metrics_path__: /rhn/metrics
```

```
# 관리되는 시스템 메트릭:
- job_name: 'mgr-clients'
  uyuni_sd_configs:
   - server: "http://server.url"
     username: "admin"
     password: "admin"
  relabel_configs:
   - source_labels: [__meta_uyuni_exporter]
     target_label: exporter
   - source_labels: [__address__]
     replacement: "No group"
     target_label: groups
   - source_labels: [__meta_uyuni_groups]
     regex: (.+)
     target_label: groups
   - source_labels: [__meta_uyuni_minion_hostname]
     target_label: hostname
   - source_labels: [__meta_uyuni_primary_fqdn]
     regex: (.+)
     target_label: hostname
   - source_labels: [hostname, __address__]
     regex: (.*);.*:(.*)
     replacement: ${1}:${2}
     target_label: __address__
   - source_labels: [__meta_uyuni_metrics_path]
     regex: (.+)
     target_label: __metrics_path__
   - source_labels: [__meta_uyuni_proxy_module]
     target_label: __param_module
   - source_labels: [__meta_uyuni_scheme]
     target_label: __scheme__
```

5. 구성 파일을 저장합니다.
6. Prometheus 서비스를 재시작합니다.

```
systemctl restart prometheus
```

Prometheus 구성 옵션에 대한 자세한 내용은 https://prometheus.io/docs/prometheus/latest/configuration/configuration/의 공식 Prometheus 설명서에서 확인할 수 있습니다.

16.3.2. Grafana 설치

모니터링 서버가 SUSE Multi-Linux Manager에서 관리하는 클라이언트인 경우 SUSE Multi-Linux Manager Web UI를 사용하여 Grafana 패키지를 설치할 수 있습니다. 그렇지 않으면 모니터링 서버에 패키지를 수동으로 다운로드하여 설치할 수 있습니다.

Grafana는 SUSE Multi-Linux Manager 프록시에서 사용할 수 없습니다.

절차: Web UI를 사용한 Grafana 설치

1. SUSE Multi-Linux Manager Web UI에서 Grafana를 설치할 시스템의 세부 사항 페이지를 열고 **수식** 탭으로 이동합니다.
2. 모니터링 수식을 활성화하려면 **Grafana** 확인란을 선택하고 [**저장**]을 클릭합니다.

3. 상단 메뉴의 **Grafana** 탭으로 이동합니다.
4. **Grafana 활성화 및 구성** 섹션에서 Grafana에 로그인하기 위해 사용할 관리자 자격 증명을 입력합니다.
5. **데이터 소스** 섹션에서 Prometheus URL 필드가 Prometheus를 실행 중인 시스템을 가리키는지 확인합니다.
6. 필요에 따라 기타 구성 옵션을 사용자 정의합니다.
7. [**수식 저장**]을 클릭합니다.
8. highstate를 적용하고 성공적으로 완료되었는지 확인합니다.
9. Grafana 인터페이스를 올바르게 로드했는지 확인합니다. 브라우저에서 3000 포트 Grafana가 설치된 서버의 URL(예: http://example.com:3000)로 이동합니다.
   - Grafana에 액세스할 수 있으려면 방화벽에서 포트 3000이 열려 있어야 합니다.

SUSE Multi-Linux Manager는 서버 자체 상태, 기본 클라이언트 모니터링 등을 위해 사전 빌드된 대시보드를 제공합니다. 수식 구성 페이지에서 프로비저닝할 대시보드를 선택할 수 있습니다.

절차: Grafana 수동 설치

1. **grafana** 패키지를 설치합니다.

```
zypper in grafana
```

2. Grafana 서비스를 활성화합니다.

```
systemctl enable --now grafana-server
```

3. 브라우저에서 3000 포트 Grafana가 설치된 서버의 URL(예: http://example.com:3000)로 이동합니다.
   - Grafana에 액세스할 수 있으려면 방화벽에서 포트 3000이 열려 있어야 합니다.
4. 로그인 페이지에서 사용자 이름과 비밀번호에 **admin**을 입력합니다.
5. [**로그인**]을 클릭합니다. 로그인에 성공하면 비밀번호를 변경하라는 메시지가 표시됩니다.
6. 메시지가 표시되면 [**확인**]을 클릭한 후 비밀번호를 변경합니다.

7. 구성 옵션이 표시되는 사이드 메뉴의 톱니바퀴 아이콘으로 커서를 이동합니다.
8. [**데이터 소스**]를 클릭합니다.
9. 지원되는 모든 데이터 소스 목록을 확인하려면 [**데이터 소스 추가**]를 클릭합니다.
10. Prometheus 데이터 소스를 선택합니다.
11. Prometheus 서버의 올바른 URL을 지정해야 합니다.
12. [**저장 및 테스트**]를 클릭합니다.
13. 대시보드를 임포트하려면 사이드 메뉴에서 [**+**] 아이콘을 클릭한 후 [**임포트**]를 클릭합니다.
14. SUSE Multi-Linux Manager 서버 개요의 경우 대시보드 ID: **17569**를 로드합니다.
15. SUSE Multi-Linux Manager 클라이언트 개요의 경우 대시보드 ID: **17570**을 로드합니다.

- 수식 모니터링에 대한 자세한 내용은 **Specialized-guides › Salt**에서 참조하십시오.
- Grafana의 수동 설치 및 구성 방법에 대한 자세한 내용은 https://grafana.com/docs에서 확인할 수 있습니다.

16.4. SUSE Multi-Linux Manager 모니터링 구성

SUSE Multi-Linux Manager 4 이상을 사용하면 서버가 Prometheus 자체 상태 메트릭을 노출하도록 설정하고 클라이언트 시스템에 엑스포트를 설치 및 구성할 수도 있습니다.

16.4.1. 서버 자체 모니터링

서버 자체 상태 메트릭에는 하드웨어, 운영 체제 및 SUSE Multi-Linux Manager 내부가 포함됩니다. 이러한 메트릭은 Prometheus 엑스포트와 결합된 Java 애플리케이션의 계측을 통해 사용할 수 있습니다.

SUSE Multi-Linux Manager 서버와 함께 제공되는 엑스포터는 다음과 같습니다.

- 노드 엑스포트: **golang-github-prometheus-node_exporter**.
  - https://github.com/prometheus/node_exporter를 참조하십시오.
- PostgreSQL 엑스포트: **prometheus-postgres_exporter**.
  - https://github.com/wrouesnel/postgres_exporter를 참조하십시오.
- JMX 엑스포트: **prometheus-jmx_exporter**.
  - https://github.com/prometheus/jmx_exporter를 참조하십시오.

SUSE Multi-Linux Manager 프록시와 함께 제공되는 엑스포트 패키지:

- 노드 엑스포트: **golang-github-prometheus-node_exporter**.
  - https://github.com/prometheus/node_exporter를 참조하십시오.
- Squid 엑스포트: **golang-github-boynux-squid_exporter**.
  - https://github.com/boynux/squid-exporter를 참조하십시오.

엑스포터는 SUSE Multi-Linux Manager 서버 및 프록시에 미리 설치되어 있지만, 각 systemd 데몬은 기본적으로 비활성화되어 있습니다.

절차: 자체 모니터링 활성화

1. SUSE Multi-Linux Manager Web UI에서 **관리자 › 관리자 구성 › 모니터링**으로 이동합니다.
2. **[서비스 활성화]**를 클릭합니다.
3. Tomcat 및 Taskomatic을 재시작합니다.
4. 9090 포트 Prometheus 서버의 URL(예: http://example.com:9090)로 이동합니다.
5. Prometheus UI에서 **상태 › 대상**으로 이동하여 **mgr-server** 그룹의 모든 엔드포인트가 작동 중인지 확인합니다.
6. Web UI와 함께 Grafana도 설치한 경우 SUSE Multi-Linux Manager 서버 대시보드의 **관리자 › 관리자 구성 › 모니터링**에서 서버 인사이트를 볼 수 있습니다.

Web UI를 사용하여 서버 자체 상태 모니터링만 활성화할 수 있습니다. Prometheus는 프록시에 대한 메트릭을 자동으로 수집하지 않습니다. 프록시에서 자체 상태 모니터링을 활성화하려면 엑스포트를 수동으로 설치 및 활성화해야 합니다.

SUSE Multi-Linux Manager 서버에서 수집되는 관련 메트릭은 다음과 같습니다.

표 5. 서버 통계(포트 80)

메트릭	유형	설명
uyuni_all_systems	게이지	전체 시스템 수
uyuni_virtual_systems	게이지	가상 시스템 수
uyuni_inactive_systems	게이지	비활성 시스템 수
uyuni_outdated_systems	게이지	오래된 패키지가 있는 시스템 수

표 6. PostgreSQL 엑스포트(9187 포트)

메트릭	유형	설명
pg_stat_database_tup_fetched	counter	쿼리에 의해 패치되는 행의 수
pg_stat_database_tup_inserted	counter	쿼리에 의해 삽입되는 행의 수
pg_stat_database_tup_updated	counter	쿼리에 의해 업데이트되는 행의 수
pg_stat_database_tup_deleted	counter	쿼리에 의해 삭제되는 행의 수
mgr_serveractions_completed	gauge	완료된 작업의 수
mgr_serveractions_failed	gauge	실패한 작업의 수
mgr_serveractions_picked_up	gauge	선택된 작업의 수
mgr_serveractions_queued	gauge	대기열에 추가된 작업의 수

표 7. JMX 엑스포트(Tomcat 포트 5556, Taskomatic 포트 5557)

메트릭	유형	설명
java_lang_Threading_ThreadCount	gauge	활성 스레드의 수
java_lang_Memory_HeapMemoryUsage_used	gauge	현재 힙 메모리 사용량

표 8. 서버 메시지 대기열(포트 80)

메트릭	유형	설명
message_queue_thread_pool_threads	counter	생성된 메시지 대기열 스레드의 수
message_queue_thread_pool_threads_active	gauge	현재 활성 메시지 대기열 스레드의 수
message_queue_thread_pool_task_count	counter	제출된 태스크의 수
message_queue_thread_pool_completed_task_count	counter	완료된 태스크의 수

표 9. Salt 대기열(포트 80)

메트릭	유형	설명
salt_queue_thread_pool_size	gauge	Salt 대기열 당 생성된 스레드의 수
salt_queue_thread_pool_active_threads	gauge	대기열 당 현재 활성 Salt 스레드의 수
salt_queue_thread_pool_task_total	counter	대기열당 제출된 태스크의 수
salt_queue_thread_pool_completed_task_total	counter	대기열당 완료된 태스크의 수

모든 salt_queue 값에는 대기열 번호를 값으로 갖는 **queue**라는 레이블이 있습니다.

표 10. Taskomatic Scheduler(포트 9800)

메트릭	유형	설명
taskomatic_scheduler_threads	counter	생성된 스케줄러 스레드의 수
taskomatic_scheduler_threads_active	gauge	현재 활성 스케줄러 스레드의 수
taskomatic_scheduler_completed_task_count	counter	완료된 태스크의 수

16.4.2. 관리되는 시스템 모니터링

Prometheus 메트릭 엑스포트는 수식을 사용하여 Salt 클라이언트에 설치 및 구성할 수 있습니다. 패키지는 SUSE Multi-Linux Manager 클라이언트 도구 채널에서 사용할 수 있으며 SUSE Multi-Linux Manager Web UI에서 직접 활성화 및 구성할 수 있습니다.

다음 엑스포트는 관리되는 시스템에 설치할 수 있습니다.

- 노드 엑스포트: **golang-github-prometheus-node_exporter**.
  - https://github.com/prometheus/node_exporter를 참조하십시오.
- PostgreSQL 엑스포트: **prometheus-postgres_exporter**.
  - https://github.com/wrouesnel/postgres_exporter를 참조하십시오.
- Apache 엑스포트: **golang-github-lusitaniae-apache_exporter**.
  - https://github.com/Lusitaniae/apache_exporter를 참조하십시오.

SL Micro에서는 Node 엑스포터와 Blackbox 엑스포터만 사용할 수 있습니다.

엑스포트를 설치 및 구성한 후에는 Prometheus를 사용하여 모니터링되는 시스템에서 메트릭 수집을 시작할 수 있습니다. 모니터링 서버를 Web UI로 구성한 경우 메트릭 수집이 자동으로 수행됩니다.

절차: 클라이언트에서 Prometheus 엑스포트 구성

1. SUSE Multi-Linux Manager Web UI에서 모니터링할 클라이언트의 세부 사항 페이지를 열고 **수식** 탭으로 이동합니다.
2. **Prometheus 엑스포트** 수식에서 **활성화됨** 확인란을 선택합니다.
3. **[저장]**을 클릭합니다.
4. **수식 › Prometheus 엑스포트** 탭으로 이동합니다.
5. 활성화할 엑스포트를 선택하고 필요에 따라 인수를 사용자 정의합니다. **주소** 필드에는 콜론이 앞에 오는 포트 번호(**:9100**) 또는 완전히 확인할 수 있는 주소(**example:9100**)를 사용할 수 있습니다.
6. **[수식 저장]**을 클릭합니다.
7. highstate를 적용합니다.

다음 포트는 방화벽에 의해 차단되지 않고 활성화되어야 합니다.

- 모니터: 9090/tcp
- 모니터링되는 시스템: 9100/tcp, 9117/tcp 및 9187/tcp

해당 그룹 내의 개별 시스템에 사용된 동일한 구성을 적용하여 시스템 그룹에 대한 모니터링 수식을 구성할 수도 있습니다.

수식 모니터링에 대한 자세한 내용은 **Specialized-guides › Salt**에서 참조하십시오.

16.4.3. Grafana 비밀번호 변경

Grafana 비밀번호를 변경하려면 Grafana 설명서에 설명된 단계를 따릅니다.

- https://grafana.com/docs/grafana/latest/administration/user-management/user-preferences/#change-your-grafana-password

Grafana 관리자 비밀번호를 분실한 경우 다음 명령을 사용하여 **root**로 재설정할 수 있습니다.

```
grafana-cli --configOverrides cfg:default.paths.data=/var/lib/grafana --homepath
/usr/share/grafana admin reset-admin-password <new_password>
```

16.5. 네트워크 경계

Prometheus는 풀 방식을 사용하여 메트릭을 가져오므로 서버는 모니터링되는 클라이언트에 TCP 연결을 설정할 수 있어야 합니다. 기본적으로 Prometheus에서 사용되는 포트입니다.

- 노드 엑스포트: 9100

- PostgreSQL 엑스포트: 9187
- Apache 엑스포트: 9117

추가적으로, Prometheus를 실행하는 호스트가 아닌 호스트에서 경고 관리자를 실행하는 경우에는 포트 9093도 열어야 합니다. 경고 관리자는 Prometheus 솔루션의 일부입니다. 경고 관리자는 Prometheus 서버 인스턴스와 같은 클라이언트 애플리케이션에서 보내는 경고를 처리합니다. 경고 관리자에 대한 자세한 내용은 https://prometheus.io/docs/alerting/latest/alertmanager/에서 확인할 수 있습니다.

클라우드 인스턴스에 설치된 클라이언트의 경우 모니터링 서버에 대한 액세스 권한이 있는 보안 그룹에 필요한 포트를 추가할 수 있습니다.

또는 엑스포트의 로컬 네트워크에 Prometheus 인스턴스를 배포하고 페더레이션을 구성할 수 있습니다. 이를 통해 기본 모니터링 서버가 로컬 Prometheus 인스턴스에서 시계열을 스크랩할 수 있습니다. 이 방법을 사용하는 경우 Prometheus API 포트인 9090 포트만 열면 됩니다.

Prometheus 페더레이션에 대한 자세한 내용은 https://prometheus.io/docs/prometheus/latest/federation/에서 확인할 수 있습니다.

네트워크 경계를 통해 요청을 프록시할 수도 있습니다. PushProx와 같은 도구는 네트워크 장벽의 양쪽에 프록시와 클라이언트를 배포하고 Prometheus가 NAT와 같은 네트워크 토폴로지에서 작동할 수 있도록 합니다.

PushProx에 대한 자세한 내용은 https://github.com/RobustPerception/PushProx에서 확인할 수 있습니다.

16.5.1. 역방향 프록시 설정

Prometheus는 풀 방식을 사용하여 메트릭을 가져오므로, 서버는 모니터링되는 클라이언트의 각 엑스포트에 대한 TCP 연결을 설정할 수 있어야 합니다. 방화벽 구성을 단순화하기 위해 엑스포트에 역방향 프록시를 사용하여 단일 포트의 모든 메트릭을 노출할 수 있습니다.

절차: 역방향 프록시를 사용하여 Prometheus 엑스포터 설치하기

1. SUSE Multi-Linux Manager Web UI에서 모니터링할 시스템의 세부 사항 페이지를 열고 **수식** 탭으로 이동합니다.
2. **Prometheus 엑스포트** 확인란을 선택하여 엑스포트 수식을 활성화하고 **[저장]**을 클릭합니다.
3. 상단 메뉴에서 **Prometheus 엑스포트** 탭으로 이동합니다.
4. **역방향 프록시 활성화** 옵션을 확인하고 유효한 역방향 프록시 포트 번호를 입력합니다.
   - 방화벽에서 포트 9999/tcp가 열려 있는지 확인합니다.
5. 필요에 따라, 다른 엑스포트를 사용자 정의합니다.
6. **[수식 저장]**을 클릭합니다.
7. highstate를 적용하고 성공적으로 완료되었는지 확인합니다.

수식 모니터링에 대한 자세한 내용은 **Specialized-guides › Salt**에서 참조하십시오.

16.6. 보안

Prometheus 서버 및 Prometheus 노드 엑스포트는 TLS 암호화 및 인증으로 엔드포인트를 보호하는 기본 제공 방법을 제공합니다. SUSE Multi-Linux Manager Web UI는 관련된 모든 구성요소의 구성을 간소화합니다. TLS 인증서는 사용자가 제공하고 배포해야 합니다. SUSE Multi-Linux Manager에서 지원하는 보안 모델은 다음과 같습니다.

- 노드 엑스포트: TLS 암호화 및 클라이언트 인증서 기반 인증
- Prometheus: TLS 암호화 및 기본 인증

사용 가능한 모든 옵션 구성에 대한 자세한 내용은 **Specialized-guides › Salt**에서 확인할 수 있습니다.

16.6.1. TLS 인증서 생성

기본적으로 SUSE Multi-Linux Manager는 모니터링 구성을 보호하기 위한 인증서를 제공하지 않습니다. 보안을 제공하기 위해 자체 서명되거나 타사 인증 기관(CA)에서 서명한 사용자 지정 인증서를 생성하거나 임포트할 수 있습니다.

이 섹션에서는 SUSE Multi-Linux Manager CA를 사용하여 자체 서명된 Prometheus 및 노드 엑스포트 미니언에 대한 클라이언트/서버 인증서를 생성하는 방법을 설명합니다.

절차: 서버/클라이언트 TLS 인증서 생성

1. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 루트 권한으로 다음 명령을 실행합니다.

   a. 인증서 파일을 생성하려면 다음 명령을 실행합니다.

   Ensure that the **set-cname** parameter is the fully qualified domain name (FQDN) of your Salt client. You can use the **set-cname** parameter multiple times if you require multiple aliases:

```
rhn-ssl-tool --gen-server --dir="/root/ssl-build" --set-country="COUNTRY" \
--set-state="STATE" --set-city="CITY" --set-org="ORGANIZATION" \
--set-org-unit="ORGANIZATION UNIT" --set-email="name@example.com" \
--set-hostname="minion.example.com" --set-cname="minion.example.com" --no
-rpm
```

   결과:

```
웹 서버의 SSL 개인 키 생성: /root/ssl-build/minion/server.key
웹 서버의 SSL 인증서 요청 생성: /root/ssl-build/minion/server.csr
웹 서버의 SSL 인증서 생성/서명: server.crt
```

   b. 서버 컨테이너에서 호스트로 **server.crt** 및 **server.key** 파일을 복사합니다.

```
mgrctl cp 서버: /root/ssl-build/minion/server.key server.key
```

```
mgrctl cp 서버: /root/ssl-build/minion/server.crt server.crt
```

c. 호스트에서 모니터링 클라이언트로 **server.crt** 및 **server.key** 파일을 복사합니다.

```
ssh minion.example.com 'mkdir /etc/ssl/mlm-server-certs'
scp /root/server.* minion.example.com:/etc/ssl/mlm-server-certs
ssh minion.example.com 'chmod go+r /etc/ssl/mlm-server-certs/server.*; ls
-la /etc/ssl/mlm-server-certs'
```

2. Salt 양식을 구성하려면 이전 단계에서 지정한 디렉토리 이름을 입력합니다.

a. formular server

```
Server Certificate /etc/ssl/mlm-server-certs/server.crt
Server Key /etc/ssl/mlm-server-certs/server.key
```

b. formular 미니언

Chapter 17. 조직

조직은 SUSE Multi-Linux Manager 내에서 사용자 액세스 및 권한을 관리하기 위한 용도로 사용됩니다.

대부분의 환경에서는 조직 한 개로 충분합니다. 그러나 더 복잡한 환경에는 여러 조직이 필요할 수 있습니다. 비즈니스 내의 각 물리적 위치 또는 다양한 비즈니스 기능에 대한 조직이 필요할 수 있습니다.

조직을 생성한 후에는 사용자를 생성하여 조직에 할당할 수 있습니다. 그런 다음 조직 수준에서 권한을 할당할 수 있고, 이는 기본적으로 조직에 할당된 모든 사용자에게 적용됩니다.

PAM 및 싱글 사인온 등 새 조직에 대한 인증 방법을 구성할 수도 있습니다. 인증에 대한 자세한 내용은 **Administration › Auth-methods**에서 확인할 수 있습니다.

조직을 만들고 관리하려면 SUSE Multi-Linux Manager 관리자로 로그인해야 합니다.

절차: 새 조직 생성

1. SUSE Multi-Linux Manager Web UI에서 **관리자 › 조직**으로 이동하고 **[조직 생성]**을 클릭합니다.
2. **조직 상자** 대화 상자에서 다음 필드를 입력합니다.
   - **조직 이름** 필드에 새 조직의 이름을 입력합니다. 이름은 3~128자 사이여야 합니다.
   - **원하는 로그인** 필드에 조직 관리자에서 사용할 로그인 이름을 입력합니다. 새 관리자 계정이어야 하며, 현재 로그인한 조직을 포함하여 기존 관리자 계정을 사용해서는 새 조직에 로그인할 수 없습니다.
   - **원하는 비밀번호** 필드에 새 조직 관리자의 비밀번호를 입력합니다. **비밀번호 확인** 필드에 비밀번호를 다시 입력하여 비밀번호를 확인합니다. 암호 강도는 암호 필드 아래에 있는 색상 막대로 표시됩니다.
   - **이메일** 필드에 새 조직 관리자의 이메일 주소를 입력합니다.
   - **이름** 필드에서 인사말을 선택하고 새 조직 관리자의 이름을 입력합니다.
   - **성** 필드에 새 조직 관리자의 성을 입력합니다.
3. **[조직 생성]**을 클릭합니다.

17.1. 조직 관리

SUSE Multi-Linux Manager Web UI에서 **관리자 › 조직**으로 이동하여 사용할 수 있는 조직 목록을 확인합니다. 관리할 조직의 이름을 클릭합니다.

관리자 › 조직 섹션에서 탭에 액세스하여 조직의 사용자, 신뢰, 구성 및 상태를 관리할 수 있습니다.

조직은 관리자만 관리할 수 있습니다. 조직을 관리하려면 변경할 조직의 올바른 관리자로 로그인해야 합니다.

17.1.1. 조직 사용자

사용자 탭으로 이동하여 조직과 연결된 모든 사용자의 목록과 해당 역할을 확인합니다. 사용자 이름을 클릭하면 **사용자** 메뉴로 이동하여 사용자를 추가, 변경 또는 삭제할 수 있습니다.

17.1.2. 신뢰할 수 있는 조직

신뢰 탭으로 이동하여 신뢰할 수 있는 조직을 추가하거나 제거합니다. 조직 간에 신뢰를 설정하면 조직 간에 컨텐트를 공유할 수 있고 한 조직에서 다른 조직으로 클라이언트를 전송할 수 있습니다.

17.1.3. 조직 구성

구성 탭으로 이동하여 조직의 구성을 관리합니다. 여기에는 스테이징된 컨텐트의 사용과 SCAP 파일의 사용이 포함됩니다.

컨텐트 스테이징에 대한 자세한 내용은 **Administration › Content-staging**에서 확인할 수 있습니다.

OpenSCAP에 대한 자세한 내용은 **Reference › Audit**에서 확인할 수 있습니다.

17.2. 상태 관리

상태 탭으로 이동하여 조직의 모든 클라이언트에 대한 Salt 상태를 관리합니다. 상태를 사용하면 전역 보안 정책을 정의하거나 모든 클라이언트에 공통 관리 사용자를 추가할 수 있습니다.

Salt 상태에 대한 자세한 내용은 **Specialized-guides › Salt**에서 확인할 수 있습니다.

17.2.1. 구성 채널 관리

조직 전체에 적용되어야 하는 구성 채널을 선택할 수 있습니다. 구성 채널은 **구성 › 채널**으로 이동하여 SUSE Multi-Linux Manager Web UI에서 생성할 수 있습니다. SUSE Multi-Linux Manager Web UI를 사용하여 조직에 구성 채널을 적용합니다.

절차: 조직에 구성 채널 적용

1. SUSE Multi-Linux Manager Web UI에서 **홈 › 내 조직 › 구성 채널**로 이동합니다.
2. 검색 기능을 사용하여 이름으로 채널을 찾습니다.
3. 적용할 채널을 확인하고 **[변경사항 저장]**을 클릭합니다. 이 작업을 수행하면 변경 사항이 데이터베이스에 저장되지만, 채널에는 적용되지 않습니다.
4. **[적용]**을 클릭하여 변경사항을 적용합니다. 이 작업을 수행하면 조직 내의 모든 클라이언트에 변경사항을 적용하는 작업이 예약됩니다.

Chapter 18. 패치 관리

이 장에는 패치 관리와 관련된 다양한 항목이 포함되어 있습니다.

18.1. 철회된 패치

벤더가 새 패치를 릴리스하면 테스트에서 확인되지 않은 일부 상황에서는 패치에 바람직하지 않은 부작용(보안, 안정성)이 있을 수 있습니다. 그러한 경우(매우 드물게) 벤더는 일반적으로 새 패치를 릴리스하며, 해당 벤더의 내부 프로세스에 따라 몇 시간 또는 며칠이 걸릴 수 있습니다.

SUSE는 권고 상태를 ("최종" 또는 "안정" 대신) "철회"로 설정하여 이러한 패치를 거의 즉시 취소하는 "철회된 패치"라는 새로운 메커니즘(2021년)을 도입했습니다.

패치의 권고 상태 특성이 "철회"로 설정되면 패치가 "철회"됩니다. 패키지는 "철회된" 패치에 속하는 경우 "철회"됩니다.

철회된 패치 또는 패키지는 SUSE Multi-Linux Manager가 설치된 시스템에 설치할 수 없습니다. 철회된 패키지를 설치하는 유일한 방법은 **zypper install**을 사용하여 수동으로 설치하고 정확한 패키지 버전을 지정하는 것입니다. 예는 다음과 같습니다.

```
zypper install vim-8.0.1568-5.14.1
```

패치 및 패키지의 철회 상태는 SUSE Multi-Linux Manager의 Web UI에 ⊗ 아이콘으로 표시됩니다. 다음 예시를 참조할 수 있습니다.

- 채널 내 패키지 목록
- 채널 내 패치 목록

시스템에 설치된 패치 또는 패키지가 철회되면 해당 시스템의 설치된 패키지 목록에 ⊗ 아이콘도 표시됩니다. SUSE Multi-Linux Manager는 이러한 패치 또는 패키지를 다운그레이드할 수 있는 방법을 제공하지 않습니다.

18.1.1. 채널 클론

복제된 채널을 사용하는 경우에는 철회된 권고 상태가 원래 채널에서 클론으로 전파되는 것에 주의해야 합니다.

벤더 채널을 조직에 복제하면 채널 패치도 복제됩니다.

벤더가 채널의 패치를 철회하고 SUSE Multi-Linux Manager가 이 채널을 동기화하면(예: 야간 작업을 통해), "철회됨" 특성은 복제된 패치로 전파되지 않으며 복제된 채널에 가입한 클라이언트에게 표시되지 않습니다. 복제된 채널에 특성을 전파하려면 다음 방법을 사용할 수 있습니다.

- 패치 동기화(메뉴:소프트웨어[관리 > 복제된 채널 > 패치 > 동기화]). 이 기능을 사용하면 복제된 채널의 패치 특성을 원본에 맞출 수 있습니다.
- 컨텐트 라이프싸이클 관리. 컨텐트 라이프싸이클 관리의 컨텍스트에서 복제된 채널에 대한 자세한 내용은 **Client-configuration › Channels**에서 확인할 수 있습니다.

18.1.2. 패치 공유

조직에서 여러 벤더 채널 클론을 생성하면 패치가 여러 번 복제되지 않고 복제된 채널 간에 공유됩니다. 결과적으로 복제된 패치를 동기화할 때(패치 동기화 기능을 사용하거나 위에서 언급한 컨텐트 라이프싸이클 관리를 사용하여) 복제된 패치를 사용하는 모든 채널이 해당 변경사항을 따르게 됩니다.

예:

1. 두 개의 컨텐트 라이프싸이클 관리 프로젝트 **prj1** 및 **prj2**를 생각해 보겠습니다.
2. 이 두 프로젝트에는 **dev** 및 **test**의 두 가지 환경이 있습니다.
3. 이러한 두 프로젝트에는 벤더 채널이 소스 채널로 설정되어 있습니다.
4. 이 시나리오의 모든 채널(복제된 채널 총 네 개)은 벤더 채널의 최신 상태에 맞춰 정렬됩니다.
5. 벤더는 소스 채널에서 패치를 철회하고 야간 작업 중에 SUSE Multi-Linux Manager에 동기화합니다.
6. 채널 네 개 모두 패치 클론을 사용하고 패치를 직접 사용하지 않으므로 이 변경사항을 볼 수 없습니다.
7. 패치를 동기화하는 즉시(이러한 두 프로젝트 중 한 개를 빌드하거나 네 개의 복제된 채널에서 패치 동기화 기능을 사용), 패치 공유로 인해 복제된 채널 **모두** 패치가 철회된 것으로 표시됩니다.

Chapter 19. SUSE Multi-Linux Manager에서 PTF 사용

SUSE는 고객에게 직접 배송되는 현재 지원되는 모든 솔루션에 대한 임시 수정 사항을 제공합니다. 이러한 PTF(프로그램 임시 수정)는 이제 리포지토리로 사용할 수 있으며 SUSE Multi-Linux Manager에서 동기화할 수 있습니다.

19.1. PTF 패키지에 대해 알아보기

PTF 패키지는 프록시 패키지를 통해 설치되며 **ptf-xxxxxx**라는 이름이 지정됩니다. 여기서 xxxxxx는 버전이 아니라 번호와 패키지 이름의 일부입니다.

이는 소프트웨어에 수정 사항을 포함하는 패키지의 정확한 버전에 따라 달라집니다. 이러한 유형의 패키지는:

- 실수로 설치할 수 없습니다(즉, zypper 업데이트에서 절대로 설치를 제안하지 않음).
- 실수로 제거할 수 없습니다(즉, 사용자가 zypper 명령 줄에서 명시적으로 지정하는 경우를 제외하고 최신 패키지 버전이 PTF 버전을 대체하지 않음).
- 최신 버전이 PTF에서 이전에 해결한 특정 문제를 해결하는 것으로 알려진 경우에만 업데이트됩니다.
- 시스템에 이미 설치된 패키지만 업데이트합니다(즉, 소프트웨어가 여러 패키지로 분할된 경우 PTF는 현재 시스템에 설치된 패키지만 교체).

패키지의 정확한 ID는 영향을 받는 서비스를 배포/재시작하는 방법에 대한 지침과 함께 지원 사례 조사 과정에서 SUSE 지원이 제공합니다.

19.2. PTF 패키지 설치

PTF 패키지는 현재 SLE 12 및 SLE 15 기반 시스템에서만 지원됩니다. 다른 버전이나 운영 체제에는 이 기능이 없으며 해당 페이지가 표시되지 않습니다.

SUSE Multi-Linux Manager을(를) 통해 PTF 채널에 액세스하려면 L3 지원팀에 요청해야 합니다.

절차: 명령 줄을 사용한 PTF 리포지토리 활성화 및 동기화

1. 콘솔에서 **mgr-sync refresh**를 입력합니다.
2. **mgr-sync list channel**을 입력하고 SCC 계정 이름과 이름이 **ptfs**로 시작하는 채널을 찾습니다. 예를 들어, **a123456-sles-15.3-ptfs-x86_64**입니다.
3. **mgr-sync add channel <label>**을 사용하여 PTF 채널을 활성화합니다.

이제 이 채널을 사용할 수 있으며 동일한 기본 채널을 사용하는 모든 시스템에 추가할 수 있습니다.

PTF 패키지는 시스템을 업데이트할 때 자동으로 선택되지 않으므로 명시적으로 설치해야 합니다. SUSE 고객 지원에서 특정 문제를 해결하기 위한 PTF 번호를 제공합니다. 이 번호로 PTF 목록에서 프록시 패키지를 식별할 수 있습니다. SUSE Multi-Linux Manager Web UI에는 설치 가능한 PTF가 있는 모든 시스템에 해당 PTF를 나열하는 페이지가 있습니다.

절차: SUSE Multi-Linux Manager Web UI를 통해 PTF 리포지토리 활성화 및 동기화

1. SUSE Multi-Linux Manager Web UI에서 **관리 › 설정 마법사 › 제품**으로 이동하여 PTF 리포지토리를 활성화할 제품을 찾습니다.
2. 제품 동기화 상태 옆에 있는 **[제품 채널 표시]**를 클릭합니다.
3. 제품에 대한 필수 및 선택 채널을 나열하는 팝업이 표시되어야 합니다.
4. 선택적 채널 목록에서 SCC 계정 이름과 이름에 **ptfs**로 시작하는 채널을 찾습니다. 예를 들어, **a123456-sles-15.3-ptfs-x86_64**입니다.
5. 채널 이름 옆의 확인란을 사용하여 채널을 선택하고 **[확인]**을 클릭하여 동기화를 예약합니다.

제품이 설치되어 있어야 제품에 옵션 채널을 추가할 수 있습니다.

절차: PTF 패키지 설치

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록**으로 이동하여 PTF를 설치할 클라이언트를 선택합니다.
2. **시스템 › 소프트웨어 › 패키지 › 소프트웨어 채널**로 이동하여 **PTF 채널**을 선택합니다.
3. **[다음]**을 클릭하고 **소프트웨어 채널 변경 확인**에서 **[확인]**을 클릭합니다.
4. 채널 할당이 완료되었는지 확인하려면 **시스템 › 이벤트 › 기록**으로 이동하여 결과를 확인합니다.
5. **시스템 › 소프트웨어 › PTF › 설치** 하위 탭으로 이동합니다.
6. 설치할 PTF 패키지를 선택합니다.
7. **[PTF 설치]**를 클릭하고 **프로그램 임시 수정(PTF) 설치 확인**에서 **[확인]**을 클릭합니다.
8. PTF 설치 결과를 확인하려면 **시스템 › 이벤트 › 기록**으로 이동합니다.

API를 사용하여 PTF를 설치해야 하는 경우, 프록시 패키지 이름과 함께 일반 **system.schedulePackageInstall** API를 사용할 수 있습니다.

19.3. PTF 설치 후

보고된 문제를 해결하는 PTF가 확인되면, 업데이트된 패키지는 추적되어 향후 유지보수 업데이트에 포함된 후 업데이트 리포지토리에 정기 유지보수 업데이트로 널리 배포됩니다.

수정이 포함된 이 정기 업데이트가 릴리스되면 PTF의 업데이트된 버전도 계정별 PTF 리포지토리로 릴리스됩니다. 업데이트된 PTF에서는 엄격한 종속성이 제거되며 업데이트를 다시 설치할 수 있도록 합니다.

PTF를 수정이 포함된 유지보수 업데이트로 교체하면 표준 패키지 업데이트 또는 패치 설치를 통해 자동으로 수행됩니다.

19.4. 패키지의 패치 버전 제거

PTF를 설치 제거하고 패키지의 패치되지 않은 버전을 시스템에 설치해야 하는 경우 간단한 패키지 제거를 통해 이를 수행할 수 없습니다. PTF 패키지는 표준 패키지 목록 페이지에서 선택할 수 없습니다.

절차: PTF 패키지 제거

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록**으로 이동하여 PTF를 제거할 클라이언트를

선택합니다.

2. **시스템 › 소프트웨어 › PTF › 목록/제거** 하위 탭으로 이동합니다.
3. 제거할 ptf 패키지를 선택합니다.
4. **[PTF 제거]**를 클릭하고 **프로그램 임시 수정(PTF) 제거 확인** 페이지에서 **[확인]**을 클릭합니다.
5. 결과를 확인하려면 **시스템 › 이벤트 › 이력**으로 이동합니다.

PTF를 제거하기 위해서는 클라이언트 시스템에 특수 버전의 **libzypp** 및 **zypper**가 설치되어 있어야 합니다. **zypper --help**를 수행하여 **removeptf**가 지원되는지 확인합니다. 이 조건이 충족되는 경우에만 **목록/제거** 탭이 표시됩니다.

API를 사용하여 PTF를 제거해야 하는 경우 일반 **system.schedulePackageRemove** API를 프록시 패키지 이름과 함께 사용할 수 있습니다.

19.5. 클라이언트에서 패키지의 패치 버전 제거

콘솔을 사용하여 클라이언트에서 직접 PTF를 제거해야 하는 경우, 특수 명령어인 **zypper removeptf**를 사용해야 합니다. 다른 모든 방법은 오류가 발생하거나 시스템에서 중요한 패키지를 제거하여 시스템을 사용할 수 없게 만드는 등의 원치 않는 동작을 유발할 수 있습니다.

Chapter 20. 보고서 생성

SUSE Multi-Linux Manager를 사용하면 사용자가 다양한 보고서를 생성할 수 있습니다. 이 보고서는 가입한 시스템, 사용자 및 조직의 인벤토리를 작성할 때 유용하게 사용할 수 있습니다. 특히, 관리 중인 시스템이 많은 경우 보고서를 사용하는 방식이 SUSE Multi-Linux Manager Web UI에서 정보를 수동으로 수집하는 방식보다 더 간단합니다.

명령줄 도구 **spacewalk-report**를 사용하여 사전 구성된 보고서를 생성할 수 있지만, **Specialized-guides › Large-deployments**의 도입으로 완전 맞춤식 보고서를 생성할 수도 있습니다. 이는 SQL 언어를 지원하는 보고 도구를 보고 데이터베이스에 연결하고 데이터를 직접 추출하여 수행할 수 있습니다. 데이터 가용성 및 구조에 대한 자세한 내용은 보고 데이터베이스 스키마 설명서에서 확인할 수 있습니다.

20.1. spacewalk-report 사용

서버 컨테이너 내에서 단계를 실행하기 전에 **mgrctl term**을 사용합니다.

보고서를 생성하려면 **spacewalk-reports** 패키지가 설치되어 있어야 합니다. **spacewalk-report** 명령을 사용하면 SUSE Multi-Linux Manager의 컨텐트, 시스템 및 사용자 리소스에 대한 보고서를 구성 및 표시할 수 있습니다.

Specialized-guides › Large-deployments의 도입으로, 이제 **spacewalk-report**는 기본적으로 보고 데이터베이스에서 데이터를 수집합니다. 자세한 내용은 **spacewalk-report** 및 보고 데이터베이스에서 확인할 수 있습니다.

다음에 대한 보고서 생성 가능:

시스템 인벤토리

SUSE Multi-Linux Manager에 등록된 모든 시스템을 나열합니다.

패치

등록된 시스템과 관련된 모든 패치를 나열합니다. 패치를 심각도별로 정렬할 수 있을 뿐만 아니라 특정 패치에 적용되는 시스템도 정렬할 수 있습니다.

사용자

등록된 모든 사용자 및 특정 사용자와 관련된 모든 시스템을 나열합니다.

보고서를 CSV 형식으로 가져오려면 서버의 명령 프롬프트에서 다음 명령을 실행합니다.

```
spacewalk-report <report_name>
```

20.2. spacewalk-report 및 보고 데이터베이스

spacewalk-report는 기본적으로 새 보고 데이터베이스를 사용하여 데이터를 추출합니다. 이는 새로 생성된 보고서가 데이터의 구조와 형식에서 약간의 차이가 있음을 의미합니다. 모든 보고서에 공통적인 차이점은 다음과 같습니다.

- 보고서 데이터는 실시간으로 변경되지 않고 예약된 작업을 실행하는 방식으로만 업데이트됩니다.
- 데이터 중복이 제거되었으며, 이전에 "다중 값"으로 간주되었던 열에 이제 **;**로 구분된 여러 값이 포함됩니다. 이는 또한 명령줄 옵션 **--multival-on-rows** 및 **--multival-separator**가 이제 기본 동작이므로 새 보고서에 더 이상 적용할 수 없음을 의미합니다.

- 허브 시나리오에서 관리 서버와 애플리케이션 데이터베이스에서 정보가 마지막으로 업데이트된 시간을 파악하기 위해 모든 보고서에 새로운 열 **mgm_id** 및 **synced_date**가 도입되었습니다.
- 모든 부울 값은 이제 **1**/**0** 값이 아닌 **True**/**False**로 표시됩니다.
- **org_id** 열이 **organization**으로 대체되었으며, 여기에는 숫자 식별자가 아닌 조직 이름이 포함됩니다.
- "서버"라는 용어가 "시스템"으로 대체되었습니다. 예를 들어, **server_id** 열이 이제 **system_id**입니다.

구체적인 보고서 변경사항은 사용할 수 있는 보고서 목록을 참조하십시오.

이 변경된 동작으로 문제가 발생하면 새 옵션 **--legacy-report**를 사용하여 이전 보고서로 폴백할 수 있으며, 이는 애플리케이션 데이터베이스에 대해 실행됩니다.

허브 보고에 대한 자세한 내용은 **Specialized-guides › Large-deployments**에서 확인할 수 있습니다.

20.3. 사용할 수 있는 보고서 목록

이 테이블은 사용할 수 있는 보고서를 보여줍니다.

표 11. spacewalk-report Reports

보고	호출 명령	설명	보고 데이터베이스 사용	구체적인 차이점
작업	**actions**	모든 작업.	예	이제 **id** 열을 **action_id**라고 합니다.
활성화 키	**activation-keys**	모든 활성화 키, 권리 유형, 채널, 구성 채널, 시스템 그룹 및 이와 관련된 패키지.	아니요	
활성화 키: 채널	**activation-keys-channels**	모든 활성화 키 및 각 키와 연결된 엔터티.	아니요	
활성화 키: 구성	**activation-keys-config**	모든 활성화 키 및 각 키와 연결된 구성 채널.	아니요	
활성화 키: 서버 그룹	**activation-keys-groups**	모든 활성화 키 및 각 키와 연결된 시스템 그룹.	아니요	
활성화 키: 패키지	**activation-keys-packages**	모든 활성화 키 및 각 키가 배포할 수 있는 패키지.	아니요	
채널 패키지	**channel-packages**	채널의 모든 패키지.	예	
채널 보고서	**channels**	지정된 채널에 대한 자세한 보고서.	예	

보고	호출 명령	설명	보고 데이터베이스 사용	구체적인 차이점
복제된 채널 보고서	**cloned-channels**	복제된 채널에 대한 자세한 보고서.	예	
구성 파일	**config-files**	파일 내용 및 파일 정보를 포함하여 모든 조직에 대한 모든 구성 파일 개정판.	아니요	
최신 구성 파일	**config-files-latest**	파일 내용 및 파일 정보를 포함하여 모든 조직에 대한 최신 구성 파일 개정판.	아니요	
사용자 정의 채널	**custom-channels**	특정 조직이 소유한 모든 채널에 대한 채널 메타데이터.	예	이제 **id** 열을 **channel_id**라고 합니다.
사용자 정의 정보	**custom-info**	클라이언트 사용자 정의 정보.	예	
채널의 패치	**errata-channels**	채널의 모든 패치.	예	
패치 세부 사항	**errata-list**	등록된 클라이언트에 영향을 미치는 모든 패치.	예	
모든 패치	**errata-list-all**	모든 패치.	아니요	
클라이언트용 패치	**errata-systems**	적용 가능한 패치 및 영향을 받는 등록된 클라이언트.	예	
호스트 게스트	**host-guests**	호스트 및 게스트 매핑.	예	
비활성 클라이언트	**inactive-systems**	비활성 클라이언트.	예	이제 필수 파라미터를 **threshold**라고 합니다.
시스템 인벤토리	**inventory**	하드웨어 및 소프트웨어 정보와 함께 서버에 등록된 클라이언트.	예	**osad_status** 열이 제거되었습니다.
킥스타트 스크립트	**kickstart-scripts**	세부 사항이 포함된 모든 킥스타트 스크립트.	아니요	
킥스타트 트리	**kickstartable-trees**	킥스타트 가능 트리.	아니요	"

보고	호출 명령	설명	보고 데이터베이스 사용	구체적인 차이점
모든 업그레이드 가능 버전	**packages-updates-all**	업그레이드가 가능한 모든 최신 패키지 버전.	예	
최신 업그레이드 버전	**packages-updates-newest**	업그레이드가 가능한 최신 패키지 버전.	예	
프록시 개요	**proxies-overview**	모든 프록시 및 각각에 등록된 클라이언트.	예	
리포지토리	**repositories**	연결된 SSL 세부 사항 및 필터가 포함된 모든 리포지토리.	아니요	
SCAP 결과	**scap-scan**	OpenSCAP **sccdf** 평가의 결과.	예	
SCAP 결과	**scap-scan-results**	다른 형식의 OpenSCAP **sccdf** 평가 결과.	예	
시스템 데이터	**splice-export**	스플라이스 통합에 필요한 클라이언트 데이터.	아니요	
시스템 통화	**system-currency**	등록된 각 클라이언트에 대해 사용 가능한 패치 수.	아니요	
시스템 추가 패키지	**system-extra-packages**	클라이언트가 구독하는 채널에서 사용할 수 없는 모든 클라이언트에 설치된 모든 패키지.	예	
시스템 그룹	**system-groups**	시스템 그룹.	예	
시스템 하드웨어	**system-hardware**	시스템 하드웨어 정보.	아니요	
시스템 그룹용 활성화 키	**system-groups-keys**	시스템 그룹에 대한 활성화 키.	아니요	
시스템 그룹의 시스템	**system-groups-systems**	시스템 그룹의 클라이언트.	예	
시스템 그룹 사용자	**system-groups-users**	권한이 있는 시스템 그룹 및 사용자.	아니요	
과거 기록: 시스템	**system-history**	각 클라이언트에 대한 이벤트 과거 기록.	예	

보고	호출 명령	설명	보고 데이터베이스 사용	구체적인 차이점
과거 기록: 채널	**system-history-channels**	채널 이벤트 과거 기록.	예	
과거 기록: 구성	**system-history-configuration**	구성 이벤트 과거 기록.	예	**created_date** 열이 제거되었습니다.
과거 기록: 권한 부여	**system-history-entitlements**	시스템 권한 부여 이벤트 기록.	예	
과거 기록: 오류	**system-history-errata**	오류 이벤트 과거 기록.	예	**created_date** 열이 제거되었습니다.
과거 기록: 킥스타트	**system-history-kickstart**	킥스타트 이벤트 과거 기록.	예	**created_date** 열이 제거되었습니다.
과거 기록: 패키지	**system-history-packages**	패키지 이벤트 과거 기록.	예	**created_date** 열이 제거되었습니다.
과거 기록: SCAP	**system-history-scap**	OpenSCAP 이벤트 과거 기록.	예	**created_date** 열이 제거되었습니다.
MD5 인증서	**system-md5-certificates**	MD5 체크섬이 포함된 인증서를 사용하는 등록된 모든 클라이언트.	아니요	
설치된 패키지	**system-packages-installed**	클라이언트에 설치된 패키지.	예	
시스템 프로파일	**system-profiles**	소프트웨어 및 시스템 그룹 정보와 함께 서버에 등록된 모든 클라이언트.	아니요	
사용자	**users**	SUSE Multi-Linux Manager에 등록된 모든 사용자.	예	**organization_id** 열이 제거되었습니다.
MD5 사용자	**users-md5**	세부 사항 및 역할과 함께 MD5로 암호화된 비밀번호를 사용하는 모든 조직의 모든 사용자.	예	**organization_id** 열이 제거되었습니다.
관리되는 시스템	**users-systems**	개별 사용자가 관리할 수 있는 클라이언트.	예	**organization_id** 열이 제거되었습니다.

개별 보고서에 대한 자세한 내용은 **--info** 또는 **--list-fields-info** 옵션 및 보고서 이름을 사용하여 **spacewalk-report**를 실행하여 확인할 수 있습니다. 그러면 보고서에서 가능한 필드의 설명과 목록이 표시됩니다.

프로그램 호출 및 옵션에 대한 자세한 내용은 **spacewalk-report(8)** 사용자 지정 페이지 및 **spacewalk-report** 명령의

--help 파라미터를 참조하십시오.

Chapter 21. 보안

21.1. 감사

SUSE Multi-Linux Manager에서는 일련의 감사 작업을 통해 고객을 추적할 수 있습니다. 클라이언트에 모든 CVE(공개 보안 패치)가 적용되어 최신 상태인지 확인하고, 구독 일치를 수행하며, OpenSCAP를 사용하여 사양 준수 여부를 확인할 수 있습니다.

SUSE Multi-Linux Manager Web UI에서 **감사**로 이동하여 감사 작업을 수행합니다.

21.1.1. CVE 감사

CVE(공통 취약점 및 노출)는 공개적으로 알려진 보안 취약점에 대한 수정입니다.

사용할 수 있게 된 즉시 CVE를 클라이언트에 적용해야 합니다.

각 CVE에는 식별 번호, 취약성에 대한 설명 및 추가 정보에 대한 링크가 포함되어 있습니다. CVE 식별 번호의 형식은 **CVE-YEAR-XXXX**입니다.

SUSE Multi-Linux Manager Web UI에서 **감사 › CVE 감사**로 이동하여 모든 클라이언트 목록과 현재 패치 상태를 확인합니다.

기본적으로 패치 데이터는 매일 23:00시에 업데이트됩니다. 데이터를 새로 고쳐 최신 패치가 있는지 확인한 후 CVE 감사를 시작하는 것이 좋습니다.

절차: 패치 데이터 업데이트

1. SUSE Multi-Linux Manager Web UI에서 **관리자 › 작업 스케줄**으로 이동하여 **cve-server-channels-default** 일정을 선택합니다.
2. **[cve-server-channels-bunch]**를 클릭합니다.
3. **[단일 실행 일정]**을 클릭하여 작업을 예약합니다. 작업을 완료한 후 CVE 감사를 계속 진행합니다.

절차: 패치 상태 확인

1. SUSE Multi-Linux Manager Web UI에서 **감사 › CVE 감사**로 이동합니다.
2. 특정 CVE의 패치 상태를 확인하려면 **CVE 번호** 필드에 CVE 식별자를 입력합니다.
3. 찾을 패치 상태를 선택하거나 모든 상태를 확인하려면 모든 상태를 선택된 상태로 유지합니다.
4. **[서버 감사]**를 클릭하여 모든 시스템을 확인하거나 **[이미지 감사]**를 클릭하여 모든 이미지를 확인합니다.

이 페이지에서 사용되는 패치 상태 아이콘에 대한 자세한 내용은 **Reference › Audit**에서 확인할 수 있습니다.

동작 열에서는 각 시스템에 대한 취약점을 해결하기 위해 수행해야 하는 작업에 대한 정보를 제공합니다. 해당하는 경우 후보 채널 또는 패치 목록도 제공됩니다. 추가적인 일괄 처리를 위해 **시스템 세트**에 시스템을 할당할 수도 있습니다.

SUSE Multi-Linux Manager API를 사용하여 클라이언트의 패치 상태를 확인할 수 있습니다. **audit.listSystemsByPatchStatus** API 메소드를 사용하면 됩니다. 이 방법에 대한 자세한 내용은 SUSE Multi-Linux Manager API 가이드에서 확인할 수 있습니다.

21.1.2. OVAL

채널 데이터에서 CVE 정보를 검색하는 기능 외에도 SUSE Multi-Linux Manager에는 이제 OVAL 파일에서 CVE 세부 사항을 가져오는 실험적 기능이 포함되어 있습니다. 이 기능은 현재 **기술 미리보기**로 간주됩니다.

사용자가 이 기능을 실험하고 피드백을 공유하는 것은 권장됩니다. 그러나 테스트 환경에서 철저하게 테스트하지 않고 프로덕션 환경에서 사용하는 것은 아직 권장되지 않습니다.

CVE 감사 작업에서는 두 가지 주요 데이터 소스인 채널과 OVAL(개방형 취약성 및 평가 언어)을 활용합니다. 이 두 소스는 각각 다른 용도로 CVE 감사를 수행하기 위한 메타데이터를 제공합니다.

채널

채널에는 패치 등 업데이트된 소프트웨어 패키지가 포함되어 있으며, 취약점을 해결하는 데 필요한 필수 패치에 대한 정보를 제공합니다.

OVAL(기술 미리보기)

반면에 OVAL 데이터는 취약점 자체에 대한 정보와 시스템이 CVE에 취약하도록 유발하는 패키지를 제공합니다.

채널 데이터만 사용하여 CVE 감사를 수행하는 것도 가능하지만, OVAL 데이터를 동기화하면 특히 제로데이 취약점이나 부분 패치 취약점과 관련된 경우 결과의 정확성을 향상할 수 있습니다.

OVAL 데이터는 채널 데이터보다 훨씬 가볍습니다. 예를 들어, openSUSE Leap 15.4의 OVAL 데이터는 약 50MB입니다.

OVAL 데이터만 동기화한 경우 이미 CVE 감사를 수행하여 시스템이 CVE에 취약한지 여부를 확인할 수 있지만, 패치는 채널에서 제공되는 것이므로 적용할 수 없습니다.

OVAL 기능의 주요 특징은 다음과 같습니다.

- **Disabled by default**: The feature is turned off by default and must be explicitly enabled by the user by updating the configuration file **rhn.conf** and restarting relevant services.
- **Reversible**: If any issues arise, users can revert back to the standard channel-based CVE audit.
- **Performance considerations**: While initial testing has been conducted, there are still concerns regarding performance, and further optimizations may be needed.
- 기본적으로 OVAL 데이터는 매일 23:00시에 업데이트됩니다. 데이터를 새로 고쳐 최신 취약성이 있는지 확인한 후 CVE 감사를 시작하는 것이 좋습니다.

절차: OVAL 데이터 지원 활성화

1. 컨테이너의 **/etc/rhn/rhn.conf** 파일에서 다음 설정을 추가하거나 수정:

```
java.cve_audit.enable_oval_metadata=true
```

2. Tomcat 및 Taskomatic 서비스 다시 시작:

```
systemctl restart tomcat taskomatic
```

문제가 발생하여 기본 동작으로 되돌려야 하는 경우 다음을 설정하여 기능을 비활성화하십시오.

절차: OVAL 데이터 지원 비활성화

1. **rhn.conf** 파일에서 다음 설정을 추가하거나 수정:

```
java.cve_audit.enable_oval_metadata=false
```

2. Tomcat 및 Taskomatic 서비스 다시 시작:

```
systemctl restart tomcat taskomatic
```

절차: OVAL 데이터 업데이트

1. SUSE Multi-Linux Manager Web UI에서 **관리자 › 작업 스케줄**으로 이동하여 **oval-data-sync-default** 일정을 선택합니다.
2. **[oval-data-sync-bunch]**를 클릭합니다.
3. **[단일 실행 일정]**을 클릭하여 작업을 예약합니다.

작업을 완료한 후 CVE 감사를 계속 진행합니다.

21.1.2.1. CPE 수집

특정 클라이언트에 어떤 취약점이 있는지 정확하게 파악하려면 해당 클라이언트가 사용하는 운영 체제 제품을 식별해야 합니다. 이를 위해 클라이언트의 CPE(Common Platform Enumeration)를 salt 그레인으로 수집한 다음 데이터베이스에 저장합니다.

새로 등록된 클라이언트의 CPE는 자동으로 수집되어 데이터베이스에 저장됩니다. 하지만 기존 클라이언트의 경우 **패키지 목록 업데이트** 작업을 한 번 이상 실행해야 합니다.

절차: 패키지 목록 업데이트

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록 › 모두**로 이동하여 클라이언트를 선택합니다.
2. **소프트웨어** 탭으로 이동하여 **패키지** 하위 탭으로 이동합니다.
3. **[패키지 목록 업데이트]**를 클릭하여 패키지를 업데이트하고 클라이언트의 CPE를 수집합니다.

21.1.2.2. OVAL 소스

OVAL 데이터의 무결성과 최신성을 보장하기 위해 SUSE Multi-Linux Manager은(는) 모든 제품의 공식 관리자로부터 OVAL 데이터를 독점적으로 사용합니다. OVAL 데이터 소스의 목록은 아래에서 확인할 수 있습니다.

표 12. OVAL 소스

제품	소스 URL
openSUSE Leap	https://ftp.suse.com/pub/projects/security/oval
openSUSE Leap Micro	
SUSE Linux Enterprise Server	
SUSE Linux Enterprise Desktop	
SUSE Linux Enterprise Micro	
RedHat Enterprise Linux	https://www.redhat.com/security/data/oval/v2
Debian	https://www.debian.org/security/oval
Ubuntu	https://security-metadata.canonical.com/oval

OVAL 메타데이터는 일부 클라이언트, 즉 openSUSE Leap, SUSE 엔터프라이즈 제품, RHEL, Debian 또는 Ubuntu를 사용하는 클라이언트의 하위 세트에 대해서만 CVE 감사에서 사용됩니다. 그 이유는 다른 제품에 대한 OVAL 취약성 정의 메타데이터가 없기 때문입니다.

21.1.3. CVE 상태

클라이언트의 CVE 상태는 일반적으로 **영향을 받음**, **영향을 받지 않음** 또는 **패치됨**입니다. 이러한 상태는 SUSE Multi-Linux Manager에서 사용할 수 있는 정보만으로 결정됩니다.

SUSE Multi-Linux Manager 내에서 이러한 정의가 적용되는 대상은 다음과 같습니다.

특정 취약점의 영향을 받는 시스템

취약점으로 표시된 관련 패치의 패키지 버전보다 낮은 버전의 패키지가 설치된 시스템입니다.

특정 취약점의 영향을 받지 않는 시스템

취약점으로 표시된 관련 패치에도 있는 설치된 패키지가 설치되지 않은 시스템입니다.

특정 취약점에 대해 패치된 시스템

취약점으로 표시된 관련 패치와 동일한 패키지 버전 이상의 패키지가 설치된 시스템입니다.

관련 패치

관련 채널에서 SUSE Multi-Linux Manager가 알려진 패치입니다.

관련 채널

SUSE Multi-Linux Manager에서 관리하는 채널로, 시스템에 할당된 채널, 시스템에 할당된 복제된 채널의 원본, 시스템에 설치된 제품에 연결된 채널 또는 시스템에 대한 과거 또는 미래의 서비스 팩 채널입니다.

SUSE Multi-Linux Manager 내에서 사용된 정의로 인해 일부 상황에서는 CVE 감사 결과가 정확하지 않을 수 있습니다. 예를 들어, 관리되지 않는 채널, 관리되지 않는 패키지 또는 미준수 시스템은 잘못 보고할 수 있습니다.

21.2. 마스터 검증 지문에 클라이언트 설정

보안 수준이 높은 네트워크 구성에서는 Salt 클라이언트가 특정 마스터에 연결하고 있는지 확인해야 하는 경우가 있습니다. 클라이언트에서 마스터로 유효성 검사를 설정하려면 먼저 Salt 미니언 구성 파일에 마스터의 지문을 입력합니다.

- 클라이언트에서 클래식 Salt 미니언을 사용하는 경우 **/etc/salt/minion.d/custom.conf** 또는
- 클라이언트에서 Salt 번들을 사용하는 경우 **/etc/venv-salt-minion/minion.d/custom.conf**

그리고 다음 절차를 따릅니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

절차: 마스터의 지문을 클라이언트에 추가

1. 마스터의 명령 프롬프트에서 루트 권한으로 다음 명령을 사용하여 **master.pub** 지문을 찾습니다.

```
salt-key -F master
```

클라이언트에서 **/etc/salt/minion.d/custom.conf** 또는 **/etc/venv-salt-minion/minion.d/custom.conf** 구성 파일을 엽니다. 이 라인을 추가하여 예제 지문을 대체할 마스터의 지문을 입력합니다.

```
master_finger: 'ba:30:65:2a:d6:9e:20:4f:d8:b2:f3:a7:d4:65:11:13'
```

2. 서비스를 다시 시작합니다. salt-minion의 경우 다음을 실행합니다.

```
systemctl restart salt-minion
```

3. 또는 venv-salt-minion의 경우 다음을 실행합니다.

```
systemctl restart venv-salt-minion
```

Salt 번들에 대한 자세한 내용은 **Client-configuration › Contact-methods-saltbundle**에서 확인할 수 있습니다.

클라이언트에서 보안 구성에 대한 내용은 https://docs.saltproject.io/en/latest/ref/configuration/minion.html에서 확인할 수 있습니다.

21.3. 소스 패키지 미러링

로컬에서 자체 패키지를 빌드하거나 법적인 이유로 패키지에 대한 소스 코드가 필요한 경우 SUSE Multi-Linux Manager 서버에서 소스 패키지를 미러링할 수 있습니다.

미러링 소스 패키지는 상당한 양의 디스크 공간을 사용할 수 있습니다.

서버 컨테이너 내에서 단계를 실행하기 전에 **mgrctl term**을 사용합니다.

절차: 소스 패키지 미러링

1. **/etc/rhn/rhn.conf** 구성 파일을 열고 다음 라인을 추가합니다.

```
server.sync_source_packages = 1
```

2. Spacewalk 서비스를 다시 시작하여 변경 사항을 적용합니다.

```
mgradm restart
```

현재 이 기능은 모든 리포지토리에 대해 전역적으로만 활성화할 수 있습니다. 미러링을 수행하기 위해 개별 리포지토리를 선택할 수 없습니다.

이 기능이 활성화되면 다음 번 리포지토리 동기화 후 SUSE Multi-Linux Manager Web UI에서 소스 패키지를 사용할 수 있습니다. 이는 바이너리 패키지의 소스로 표시되며 Web UI에서 직접 다운로드할 수 있습니다. 소스 패키지는 Web UI를 사용하여 클라이언트에 설치할 수 없습니다.

21.4. OpenSCAP를 사용한 시스템 보안

SUSE Multi-Linux Manager는 OpenSCAP를 사용하여 클라이언트에 대한 감사를 수행합니다. 모든 클라이언트에 대한 규정 준수 검사를 예약 및 확인할 수 있습니다.

21.4.1. SCAP 소개

SCAP(Security Content Automation Protocol)는 커뮤니티 아이디어에서 파생된 상호 운용이 가능한 사양의 종합으로, 엔터프라이즈 시스템의 시스템 보안을 유지하기 위해 NIST(National Institute of Standards and Technology)에서 유지보수하는 사양군입니다.

SCAP는 시스템 보안을 유지하기 위한 표준화된 접근 방식을 제공하기 위해 만들어졌으며, 사용되는 표준은 커뮤니티 및 엔터프라이즈 비즈니스의 니즈를 충족하기 위해 지속적으로 변경됩니다. 일관적이고 반복 가능한 개정 작업 흐름을 제공하기 위해 신규 사양은 NIST의 SCAP 릴리스 주기에 의해 관리됩니다. 자세한 내용은 다음에서 확인할 수 있습니다.

- https://csrc.nist.gov/projects/security-content-automation-protocol
- https://www.open-scap.org/features/standards/
- https://ncp.nist.gov/repository?scap

SUSE Multi-Linux Manager는 OpenSCAP를 사용하여 SCAP 사양을 구현합니다. OpenSCAP는 XCCDF(Extensible Configuration Checklist Description Format)를 활용하는 감사 도구입니다. XCCDF는 검사 목록의 내용을 나타내는 표준 방식이며 보안 검사 목록을 정의합니다. 또한, CPE(Common Platform Enumeration), CCE(Common Configuration Enumeration), OVAL(Open Vulnerability and Assessment Language)과 같은 다른 사양과 결합하여 SCAP 인증 제품에서 처리할 수 있는 SCAP로 표시되는 검사 목록을 생성합니다.

OpenSCAP는 SUSE 보안 팀에서 생성한 컨텐트를 사용하여 패치가 있는지 확인합니다. OpenSCAP은 시스템 보안 구성 설정을 확인하고 표준 및 사양에 기반한 규칙을 활용하여 시스템의 손상 징후를 검사합니다. SUSE 보안 팀에 대한 자세한 내용은 https://www.suse.com/support/security를 참조하십시오.

21.4.2. SCAP 스캔을 위해 클라이언트 준비

시작하기 전, SCAP 스캔을 위해 클라이언트 시스템을 준비해야 합니다.

SSH 연락 방법을 사용하는 Salt 클라이언트에서는 OpenSCAP 감사를 사용할 수 없습니다.

스캔 클라이언트는 스캔이 수행 중인 클라이언트에서 대규모의 메모리와 컴퓨팅 파워를 소모할 수 있습니다. Red Hat 클라이언트의 경우 스캔할 각 클라이언트에서 최소 2 GB의 RAM을 사용할 수 있어야 합니다.

시작하기 전에 클라이언트에 OpenSCAP 스캐너와 SCAP Security Guide(컨텐트) 패키지를 설치합니다. 운영 체제에 따라 이러한 패키지는 기본 운영 체제 또는 SUSE Multi-Linux Manager 클라이언트 도구에 포함되어 있습니다.

아래 테이블에는 필요한 패키지가 나열되어 있습니다.

표 13. OpenSCAP 패키지

운영 체제	스캐너	컨텐트
SLES	openscap-utils	scap-security-guide
openSUSE	openscap-utils	scap-security-guide
RHEL	openscap-utils	scap-security-guide-redhat
CentOS	openscap-utils	scap-security-guide-redhat
Oracle Linux	openscap-utils	scap-security-guide-redhat
Ubuntu	libopenscap8	scap-security-guide-ubuntu
Debian	libopenscap8	scap-security-guide-debian

RHEL 7 및 호환 시스템은 오래된 컨텐트가 포함된 **scap-security-guide** 패키지를 제공합니다. SUSE Multi-Linux Manager 클라이언트 도구에 포함된 **scap-security-guide-redhat** 패키지를 사용하는 것이 좋습니다.

SUSE는 다양한 openscap 프로파일에 대한 **scap-security-guide** 패키지를 제공합니다. **scap-security-guide**의 현재 버전에서 SUSE가 지원하는 프로파일:

- SUSE Linux Enterprise Server 12 및 15용 DISA STIG 프로파일
- SUSE Linux Enterprise Server 12 및 15용 ANSSI-BP-028 프로파일
- SUSE Linux Enterprise Server 12 및 15용 PCI-DSS 프로파일
- SUSE Linux Enterprise Server 15용 HIPAA 프로파일
- SAP Applications 15용 SUSE Linux Enterprise Server의 공용 클라우드 이미지 강화
- SUSE Linux Enterprise 15용 공용 클라우드 강화
- SLE 12 및 15용 표준 시스템 보안 프로파일

Other profiles, such as the CIS profile, are community supplied and not officially supported by SUSE.

- CIS profile for SUSE Linux Enterprise Server 12 and 15 (unsupported)

비SUSE 운영 체제의 경우 포함된 프로파일은 커뮤니티에서 제공하는 프로파일입니다. SUSE에서는 공식적으로 지원하지 않습니다.

21.4.3. OpenSCAP 컨텐트 파일

OpenSCAP는 SCAP 컨텐트 파일을 사용하여 테스트 규칙을 정의합니다. 이러한 컨텐트 파일은 XCCDF 또는 OVAL 표준을 기반으로 생성됩니다. SCAP 보안 가이드뿐만 아니라 공개적으로 사용할 수 있는 컨텐트 파일을 다운로드하고 요구사항에 따라 사용자 정의할 수 있습니다. 그리고 기본 컨텐트 파일 템플릿용 SCAP 보안 가이드 패키지를 설치할 수 있습니다. 또는 XCCDF 또는 OVAL에 익숙한 경우 자체 컨텐트 파일을 생성할 수 있습니다.

템플릿을 사용하여 SCAP 컨텐트 파일을 생성하는 것이 권장됩니다. 자체 사용자 정의 컨텐트 파일을 생성 및 사용하는 경우 자신이 모든 위험을 감수해야 합니다. 사용자 정의 컨텐트 파일을 사용하여 시스템이 손상된 경우 SUSE에서 지원하지 않을 수 있습니다.

컨텐트 파일을 생성한 후에는 해당 파일을 클라이언트로 전송해야 합니다. 물리적 저장 매체를 사용하거나 Salt(예: salt-cp 또는 Salt File Server), **ftp** 또는 **scp**를 사용하여 네트워크를 통해 다른 파일을 이동할 때와 동일한 방식으로 이 작업을 수행할 수 있습니다.

SUSE Multi-Linux Manager로 관리하는 클라이언트에 컨텐트 파일을 배포하는 패키지를 생성하는 것이 권장됩니다. 무결성을 보장하기 위해 패키지가 서명 및 검증될 수 있습니다. 자세한 내용은 **Administration › Custom-channels**에서 확인할 수 있습니다.

21.4.4. OpenSCAP 프로파일 찾기

운영 체제에 따라 다양한 OpenSCAP 컨텐트 파일 및 프로파일을 사용할 수 있습니다. 컨텐트 파일 한 개에는 프로파일이 두 개 이상 포함될 수 있습니다.

RPM 기반 운영 체제에서 다음 명령을 사용하여 사용 가능한 SCAP 파일의 위치를 확인합니다.

```
rpm -ql <scap-security-guide-package-name-from-table>
```

DEB 기반 운영 체제에서 다음 명령을 사용하여 사용 가능한 SCAP 파일의 위치를 확인합니다.

```
dpkg -L <scap-security-guide-package-name-from-table>
```

요구사항에 적합한 SCAP 컨텐트 파일 한 개를 찾은 경우 클라이언트에서 사용 가능한 프로필을 나열합니다.

```
oscap info ./usr/share/xml/scap/ssg/content/ssg-sle15-ds.xml
Document type: Source Data Stream
Imported: 2021-03-24T18:14:45

스트림: scap_org.open-scap_datastream_from_xccdf_ssg-sle15-xccdf-1.2.xml
생성됨: (null)
버전: 1.2
체크리스트:
        Ref-Id: scap_org.open-scap_cref_ssg-sle15-xccdf-1.2.xml
                상태: draft
                생성됨: 2021-03-24
                해결됨: true
```

```
                프로파일:
                        제목: CIS SUSE Linux Enterprise 15 Benchmark
                                Id: xccdf_org.ssgproject.content_profile_cis
                        제목: Standard System Security Profile for SUSE Linux Enterprise
15
                                Id: xccdf_org.ssgproject.content_profile_standard
                        제목: DISA STIG for SUSE Linux Enterprise 15
                                Id: xccdf_org.ssgproject.content_profile_stig
                참조된 확인 파일:
                        ssg-sle15-oval.xml
                                system: http://oval.mitre.org/XMLSchema/oval-definitions-5
                        ssg-sle15-ocil.xml
                                system: http://scap.nist.gov/schema/ocil/2

https://ftp.suse.com/pub/projects/security/oval/suse.linux.enterprise.15.xml
                                system: http://oval.mitre.org/XMLSchema/oval-definitions-5
확인:
        Ref-Id: scap_org.open-scap_cref_ssg-sle15-oval.xml
        Ref-Id: scap_org.open-scap_cref_ssg-sle15-ocil.xml
        Ref-Id: scap_org.open-scap_cref_ssg-sle15-cpe-oval.xml
사전:
        Ref-Id: scap_org.open-scap_cref_ssg-sle15-cpe-dictionary.xml
```

스캔을 수행하기 위한 파일 경로 및 프로파일을 기록해 둡니다.

21.4.5. 감사 스캔 수행

컨텐트 파일을 설치 또는 전송하면 감사 스캔을 수행할 수 있습니다. SUSE Multi-Linux Manager Web UI를 사용하여 감사 스캔을 트리거할 수 있습니다. SUSE Multi-Linux Manager API를 사용하여 정기 검사를 예약할 수도 있습니다.

절차: Web UI에서 감사 스캔 실행

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록**으로 이동하여 스캔할 클라이언트를 선택합니다.
2. **감사** 탭으로 이동하여 **일정** 하위 탭으로 이동합니다.
3. **XCCDF 문서 경로** 필드에 클라이언트에서 사용할 SCAP 템플릿 및 프로파일에 대한 파라미터를 입력합니다. 예는 다음과 같습니다.
   - **Command**: /usr/bin/oscap xccdf eval
   - **Command-line arguments**: --profile xccdf_org.ssgproject.content_profile_stig
   - **Path to XCCDF document**: /usr/share/xml/scap/ssg/content/ssg-sle15-ds.xml

     --fetch-remote-resources 파라미터를 사용하면 대량의 RAM이 필요합니다. 또한, **file_recv_max_size** 값을 늘려야 할 수도 있습니다.

4. 스캔은 클라이언트의 다음번에 예약된 동기화 시 실행됩니다.

XCCDF 컨텐트 파일은 유효성을 검사한 후 원격 시스템에서 실행됩니다. 컨텐트 파일에 잘못된 인수가 포함되면 테스트가 실패합니다.

절차: API에서 감사 스캔 실행

1. 시작하기 전, 스캔할 클라이언트에 Python 및 XML-RPC 라이브러리가 설치되어 있는지 확인합니다.
2. 기존 스크립트를 선택하거나 **system.scap.scheduleXccdfScan**을 통해 시스템 검사를 예약하기 위한 스크립트를 생성합니다. 예는 다음과 같습니다.

```
#!/usr/bin/python3
import xmlrpc.client
client = xmlrpc.client.ServerProxy('https://server.example.com/rpc/api')
key = client.auth.login('username', 'password')
client.system.scap.scheduleXccdfScan(key, <1000010001>,
    '<path_to_xccdf_file.xml>',
    '--profile <profile_name>')
client.auth.logout(session_key)
```

이 예에서:

- **<1000010001>**은 시스템 ID(sid)입니다.
- **<path_to_xccdf_file.xml>**은 클라이언트의 컨텐트 파일 위치에 대한 경로입니다. 예를 들어, **/usr/share/xml/scap/ssg/content/ssg-sle15-ds.xml**입니다.
- **<profile_name>**은 **oscap** 명령에 대한 추가 인수입니다. 예를 들어, **united_states_government_configuration_baseline**(USGCB)을 사용합니다.

3. 명령 프롬프트에서, 스캔할 클라이언트의 스크립트를 실행합니다.

21.4.6. 스캔 결과

Information about the scans you have run is in the SUSE Multi-Linux Manager Web UI. Navigate to **Audit › OpenSCAP › All Scans** for a table of results. For more information about the data in this table, see **Reference › Audit**.

스캔에 대한 자세한 정보를 사용하려면 클라이언트에서 활성화해야 합니다. SUSE Multi-Linux Manager Web UI에서 **관리자 › 조직**으로 이동하고 클라이언트가 속한 조직을 클릭합니다. **구성** 탭으로 이동하여 **자세한 SCAP 파일 업로드 활성화** 옵션을 선택합니다. 활성화하면, 모든 스캔에서 자세한 정보가 포함된 추가 HTML 파일이 생성됩니다. 결과에는 다음과 유사한 추가 라인이 표시됩니다.

```
자세한 결과: xccdf-report.html xccdf-results.xml scap-yast2sec-oval.xml.result.xml
```

명령줄에서 스캔 정보를 검색하려면 **spacewalk-report** 명령을 사용합니다.

```
spacewalk-report system-history-scap
spacewalk-report scap-scan
spacewalk-report scap-scan-results
```

SUSE Multi-Linux Manager API를 사용하여 **system.scap** 핸들러로 결과를 확인할 수도 있습니다.

21.4.7. 수정

클라이언트 시스템을 강화하기 위해 수정 Bash 스크립트와 Ansible 플레이북이 동일한 SCAP 보안 가이드 패키지에 제공됩니다. 예는 다음과 같습니다.

목록 5. bash 스트립트

```
/usr/share/scap-security-guide/bash/sle15-script-cis.sh
/usr/share/scap-security-guide/bash/sle15-script-standard.sh
/usr/share/scap-security-guide/bash/sle15-script-stig.sh
```

목록 6. Ansible 플레이북

```
/usr/share/scap-security-guide/ansible/sle15-playbook-cis.yml
/usr/share/scap-security-guide/ansible/sle15-playbook-standard.yml
/usr/share/scap-security-guide/ansible/sle15-playbook-stig.yml
```

클라이언트 시스템에서 Ansible을 활성화한 후 원격 명령을 사용하거나 Ansible을 사용하여 실행할 수 있습니다.

21.4.7.1. Bash 스크립트를 사용한 수정 실행

모든 대상 시스템에 **scap-security-guide** 패키지를 설치합니다. 자세한 내용은 **Administration › Ansible-setup-control-node**에서 확인할 수 있습니다.

패키지, 채널 및 스크립트는 각 운영 체제 및 배포에 따라 다릅니다. 해당 예는 예시 수정 Bash 스크립트 섹션에서 확인할 수 있습니다.

21.4.7.1.1. 단일 시스템에서 원격 명령으로 Bash 스크립트 실행

단일 시스템에서 Bash 스크립트를 원격 명령으로 실행합니다.

1. **시스템 › 개요** 탭에서 인스턴스를 선택합니다. 그런 다음 **세부 사항 › 원격 명령**에서 다음과 같이 Bash 스크립트를 작성합니다.

```
#!/bin/bash
chmod +x -R /usr/share/scap-security-guide/bash
/usr/share/scap-security-guide/bash/sle15-script-stig.sh
```

2. **[일정]**을 클릭합니다.

폴더 및 스크립트 이름은 배포판과 버전 사이에서 변경됩니다. 예제는 예시 수정 Bash 스크립트 섹션에서 확인할 수 있습니다.

21.4.7.1.2. 여러 시스템에서 시스템 세트 관리자를 사용하여 bash 스크립트 실행

한 번에 여러 시스템에서 원격 명령으로 Bash 스크립트를 실행합니다.

1. 시스템 그룹이 생성되면 **시스템 그룹**을 클릭하고 테이블에서 **SSM에서 사용**을 선택합니다.
2. **시스템 세트 관리자**의 **기타 › 원격 명령**에서 다음과 같이 Bash 스크립트를 작성합니다.

```
#!/bin/bash
chmod +x -R /usr/share/scap-security-guide/bash
/usr/share/scap-security-guide/bash/sle15-script-stig.sh
```

3. **[일정]**을 클릭합니다.

21.4.7.2. 예시 수정 Bash 스크립트

21.4.7.2.1. SUSE Linux Enterprise openSUSE 및 변형

예시 SUSE Linux Enterprise 및 openSUSE 스크립트 데이터.

패키지

scap-security-guide

채널

- SLE12: SLES12 업데이트
- SLE15: SLES15 모듈 Basesystem 업데이트

Bash 스크립트 디렉토리

/usr/share/scap-security-guide/bash/

Bash 스트립트

```
opensuse-script-standard.sh
sle12-script-standard.sh
sle12-script-stig.sh
sle15-script-cis.sh
sle15-script-standard.sh
sle15-script-stig.sh
```

21.4.7.2.2. Red Hat Enterprise Linux 및 CentOS Bash 스크립트 데이터

예시 Red Hat Enterprise Linux 및 CentOS 스크립트 데이터.

centos7-updates의 **scap-security-guide**에는 Red Hat Enterprise Linux 스크립트만 포함되어 있습니다.

패키지

scap-security-guide-redhat

채널

- SUSE Manager 도구

Bash 스크립트 디렉토리

/usr/share/scap-security-guide/bash/

Bash 스트립트

```
centos7-script-pci-dss.sh
centos7-script-standard.sh
centos8-script-pci-dss.sh
centos8-script-standard.sh
fedora-script-ospp.sh
fedora-script-pci-dss.sh
fedora-script-standard.sh
ol7-script-anssi_nt28_enhanced.sh
ol7-script-anssi_nt28_high.sh
ol7-script-anssi_nt28_intermediary.sh
ol7-script-anssi_nt28_minimal.sh
ol7-script-cjis.sh
ol7-script-cui.sh
ol7-script-e8.sh
ol7-script-hipaa.sh
ol7-script-ospp.sh
ol7-script-pci-dss.sh
ol7-script-sap.sh
ol7-script-standard.sh
ol7-script-stig.sh
ol8-script-anssi_bp28_enhanced.sh
ol8-script-anssi_bp28_high.sh
ol8-script-anssi_bp28_intermediary.sh
ol8-script-anssi_bp28_minimal.sh
ol8-script-cjis.sh
ol8-script-cui.sh
ol8-script-e8.sh
ol8-script-hipaa.sh
ol8-script-ospp.sh
ol8-script-pci-dss.sh
ol8-script-standard.sh
rhel7-script-anssi_nt28_enhanced.sh
rhel7-script-anssi_nt28_high.sh
rhel7-script-anssi_nt28_intermediary.sh
rhel7-script-anssi_nt28_minimal.sh
rhel7-script-C2S.sh
rhel7-script-cis.sh
rhel7-script-cjis.sh
rhel7-script-cui.sh
rhel7-script-e8.sh
rhel7-script-hipaa.sh
rhel7-script-ncp.sh
rhel7-script-ospp.sh
rhel7-script-pci-dss.sh
rhel7-script-rhelh-stig.sh
rhel7-script-rhelh-vpp.sh
rhel7-script-rht-ccp.sh
rhel7-script-standard.sh
rhel7-script-stig_gui.sh
rhel7-script-stig.sh
rhel8-script-anssi_bp28_enhanced.sh
rhel8-script-anssi_bp28_high.sh
rhel8-script-anssi_bp28_intermediary.sh
rhel8-script-anssi_bp28_minimal.sh
rhel8-script-cis.sh
rhel8-script-cjis.sh
rhel8-script-cui.sh
rhel8-script-e8.sh
```

```
rhel8-script-hipaa.sh
rhel8-script-ism_o.sh
rhel8-script-ospp.sh
rhel8-script-pci-dss.sh
rhel8-script-rhelh-stig.sh
rhel8-script-rhelh-vpp.sh
rhel8-script-rht-ccp.sh
rhel8-script-standard.sh
rhel8-script-stig_gui.sh
rhel8-script-stig.sh
rhel9-script-pci-dss.sh
rhosp10-script-cui.sh
rhosp10-script-stig.sh
rhosp13-script-stig.sh
rhv4-script-pci-dss.sh
rhv4-script-rhvh-stig.sh
rhv4-script-rhvh-vpp.sh
sl7-script-pci-dss.sh
sl7-script-standard.sh
```

21.4.7.2.3. Ubuntu Bash 스크립트 데이터

예시 Ubuntu 스크립트 데이터.

패키지

scap-security-guide-ubuntu

채널

- SUSE Manager 도구

Bash 스크립트 디렉토리

/usr/share/scap-security-guide/

Bash 스트립트

```
ubuntu1804-script-anssi_np_nt28_average.sh
ubuntu1804-script-anssi_np_nt28_high.sh
ubuntu1804-script-anssi_np_nt28_minimal.sh
ubuntu1804-script-anssi_np_nt28_restrictive.sh
ubuntu1804-script-cis.sh
ubuntu1804-script-standard.sh
ubuntu2004-script-standard.sh
```

21.4.7.2.4. Debian Bash 스크립트 데이터

예시 Debian 스크립트 데이터.

패키지

scap-security-guide-debian

채널

- SUSE Manager 도구

Bash 스크립트 디렉토리

/usr/share/scap-security-guide/bash/

Bash 스트립트

```
# Debian 12
debian12-script-anssi_np_nt28_average.sh
debian12-script-anssi_np_nt28_high.sh
debian12-script-anssi_np_nt28_minimal.sh
debian12-script-anssi_np_nt28_restrictive.sh
debian12-script-standard.sh
```

21.5. 리포지토리 메타데이터

리포지토리 메타데이터에 서명하려면 사용자 정의 GPG 키가 필요합니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

Procedure: Generating a custom GPG key

1. 루트 사용자로 **gpg** 명령을 사용하여 새 키를 생성합니다.

```
mgrctl exec -it -- gpg --full-generate-key
```

2. 프롬프트에서 **RSA**를 2048비트 크기의 키 유형으로 선택하고, 키의 적절한 만료 날짜를 선택합니다. 새 키에 대한 세부 사항을 확인하고 **y**를 입력하여 확인합니다.
3. 프롬프트에서 키와 연결할 이름 및 이메일 주소를 입력합니다. 원하는 경우, 키를 식별하는 데 도움이 되는 설명을 추가할 수도 있습니다. 사용자 ID에 만족하면 **O**를 입력하여 확인합니다.
4. 프롬프트에서 키를 보호하기 위한 비밀번호를 입력합니다.
5. 키는 자동으로 키링에 추가되어야 합니다. 키링에 키를 나열하여 확인합니다.

```
mgrctl exec -- gpg --list-keys
```

6. **/etc/rhn/signing.conf** 구성 파일에 키링의 비밀번호를 추가합니다. 이 작업을 수행하려면 텍스트 편집기에서 파일을 열고 다음 라인을 추가합니다.

```
GPGPASS="password"
```

GPG 키 갱신에 대한 자세한 내용은 **Administration › Troubleshooting**에서 확인할 수 있습니다.

mgr-sign-metadata-ctl 명령을 사용하여 명령줄에서 메타데이터 서명을 관리할 수 있습니다.

Procedure: Enabling metadata signing

1. 사용할 키의 짧은 식별자를 알아야 합니다. 사용할 수 있는 짧은 형식의 공용 키를 나열할 수 있습니다.

```
mgrctl exec -- gpg --keyid-format short --list-keys
...
pub   rsa4096/3E7BFE0A 2019-04-02 [SC] [expires: 2029-04-01]
      A43F9EC645ED838ED3014B035CFA51BF3E7BFE0A
uid         [ultimate] SUSE Manager
sub   rsa4096/118DE7FF 2019-04-02 [E] [expires: 2029-04-01]
```

2. **mgr-sign-metadata-ctl** 명령으로 메타데이터 서명을 활성화합니다.

```
mgrctl exec -- mgr-sign-metadata-ctl enable 3E7BFE0A
OK. Found key 3E7BFE0A in keyring.
DONE. Set key 3E7BFE0A in /etc/rhn/signing.conf.
DONE. Enabled metadata signing in /etc/rhn/rhn.conf.
DONE. Exported key 3E7BFE0A to /srv/susemanager/salt/gpg/mgr-keyring.gpg.
DONE. Exported key 3E7BFE0A to /var/spacewalk/gpg/<KEY_NAME>.key.
NOTE. For the changes to become effective run:
   mgr-sign-metadata-ctl regen-metadata
```

3. 다음 명령으로 구성이 올바른지 확인합니다.

```
mgrctl exec -- mgr-sign-metadata-ctl check-config
```

4. Restart the container for the configuration changes to be detected.

```
mgradm restart
```

5. Schedule metadata regeneration to replace all metadata with new signed versions.

```
mgrctl exec -- mgr-sign-metadata-ctl regen-metadata
```

mgr-sign-metadata-ctl 명령을 사용하여 다른 작업을 수행할 수도 있습니다. 전체 목록을 보려면 **mgr-sign-metadata-ctl --help**를 사용합니다.

리포지토리 메타데이터 서명은 전역 옵션입니다. 활성화되면 서버의 모든 소프트웨어 채널에서 활성화됩니다. 이는 서버에 연결된 모든 클라이언트가 패키지를 설치하거나 업데이트하려면 새 GPG 키를 신뢰해야 함을 의미합니다.

Procedure: Importing GPG keys on clients

1. 클라이언트에 GPG 키를 배포하면 Salt 상태 작업을 수행할 수 있습니다.
2. SUSE Multi-Linux Manager Web UI를 사용하여 highstate를 적용합니다.

GPG 키 문제 해결에 대한 자세한 내용은 **Administration › Troubleshooting**에서 확인할 수 있습니다.

Chapter 22. 역할 기반 액세스 제어(RBAC)

역할 기반 액세스 제어(RBAC)는 리소스 액세스를 할당된 역할에 따라 권한이 부여된 사용자로 제한하는 보안 방법입니다. SUSE Multi-Linux Manager에서 RBAC는 사용자가 명시적인 권한이 있는 작업만 수행하고 리소스에 액세스할 수 있도록 하여 보안을 강화하고 관리를 간소화합니다.

RBAC의 핵심 원칙에는 다음이 포함됩니다.

- **Principle of Least Privilege:** Granting only the necessary access rights for users to perform their tasks.
- **Granular Control:** Providing fine-grained control over specific functionalities.
- **Separation of Duties:** Preventing a single user from having too much control over critical processes.
- **Auditability:** Allowing for clear tracking of user actions and permissions.

22.1. 주요 RBAC 개념

효율적인 RBAC 관리를 위해서는 다음 핵심 개념을 이해하는 것이 매우 중요합니다.

- **Role:** A collection of permissions defining a specific set of capabilities within SUSE Multi-Linux Manager. Roles are assigned to users, granting the user aggregated permissions.

  역할은 사용자에게 할당되어 사용자에게 통합 권한을 부여합니다.

- **Permission:** An atomic authorization to perform a specific action, access a specific web page or call a specific API endpoint within SUSE Multi-Linux Manager. In SUSE Multi-Linux Manager, permissions are represented by namespaces and their access modes.
- **User:** An individual account that interacts with SUSE Multi-Linux Manager. Users are assigned one or more roles.
- **Namespace:** A granular unit of access control organized in a tree-like structure. Most namespaces have distinct "View" or "Modify" modes.

22.2. SUSE Multi-Linux Manager의 사용자 역할

SUSE Multi-Linux Manager은(는) 사전 정의된 역할을 제공하며, 추가적인 사용자 정의 역할을 정의하는 것을 허용합니다. 선택 사항으로 다른 역할 조합으로부터 상속받을 수 있습니다.

22.2.1. 사전 정의된 역할

사전 정의 역할 및 해당 설명에 대한 전체 목록은 administration:users.pdf에서 확인할 수 있습니다.

22.2.2. 추가 역할 정의

추가 역할을 정의하려면 다음을 수행합니다.

- 기존 역할 중 권한을 상속받을 역할을 선택합니다.

- 액세스 권한을 부여할 추가 네임스페이스를 지정합니다.

22.3. 세부 액세스 관리를 위한 네임스페이스

네임스페이스는 트리 구조로 구성된 세부적인 액세스 제어를 제공합니다. 대부분의 네임스페이스에서, 네임스페이스 내 액세스는 "보기" 및 "수정" 모드로 더욱 세분화됩니다.

표 14. 예: 이미지 관리 네임스페이스 및 액세스 모드

네임스페이스	액세스 모드	설명
cm.build	수정	컨테이너 또는 Kiwi 이미지 빌드
cm.image.import	수정	등록된 이미지 저장소에서 컨테이너 이미지 임포트
cm.image.list	보기	모든 이미지 나열
cm.image.list	수정	이미지 삭제
cm.image.overview	보기	이미지 세부 정보, 패치, 패키지, 빌드 로그 및 클러스터 정보 보기
cm.image.overview	수정	이미지 검사, 재빌드, 삭제
cm.profile.details	보기	이미지 프로파일 세부 정보 보기
cm.profile.details	수정	이미지 프로파일 생성, 프로파일 세부 정보 편집
cm.profile.list	보기	모든 이미지 프로파일 나열
cm.profile.list	수정	이미지 프로파일 삭제
cm.store.details	보기	이미지 저장소 세부 정보 보기
cm.store.details	수정	이미지 저장소 생성, 저장소 세부 정보 편집
cm.store.list	보기	모든 이미지 저장소 나열
cm.store.list	수정	이미지 저장소 삭제

네임스페이스 및 해당 설명의 전체 목록은 **access.listNamespaces** API 메서드를 호출하여 확인할 수 있습니다. 요청 및 응답 형식을 포함한 자세한 내용은 SUSE Multi-Linux Manager API 문서를 참조하십시오.

22.4. RBAC 관리

현재 RBAC 역할 및 권한은 API를 통해서만 관리할 수 있습니다. 웹 UI를 통한 사용자 역할 할당은 **Administration › Users**에서 확인할 수 있습니다.

22.4.1. API를 통한 RBAC 관리

SUSE Multi-Linux Manager API는 역할, 권한 및 사용자 할당을 프로그래밍 방식으로 관리할 수 있는 메서드를 제공합니다.

22.4.1.1. 액세스 API

이러한 API 메서드는 역할 및 관련 액세스를 관리합니다.

- **listNamespaces**: SUSE Multi-Linux Manager에서 사용할 수 있는 네임스페이스, 액세스 모드 및 관련 설명을 나열합니다.
- **listPermissions**: 역할에 대해 허용되는 네임스페이스를 나열합니다.
- **listRoles**: SUSE Multi-Linux Manager의 기존 역할을 나열합니다.
- **createRole**: 새 역할을 생성하며, 선택 사항으로 기존 역할에서 권한을 복사합니다.
- **deleteRole**: 역할을 삭제합니다.
- **grantAccess**: 네임스페이스에 대한 접근 권한을 부여합니다.
- **revokeAccess**: 네임스페이스에 대한 액세스를 취소합니다.

22.4.1.2. 사용자 API

이러한 API 메서드는 사용자 역할 할당을 관리합니다.

- **listPermissions**: 사용자에 대해 유효한 권한을 나열합니다.
- **listRoles**: 사용자에게 할당된 역할을 나열합니다.
- **addRole**: 사용자에게 역할을 할당합니다.
- **removeRole**: 사용자에게서 역할을 제거합니다.

요청 형식 및 응답 형식을 포함한 자세한 API 문서는 SUSE Multi-Linux Manager API 참조를 참조하십시오.

22.5. RBAC 모범 사례

이러한 모범 사례를 준수하면 효율적이고 관리하기 쉬운 보안 RBAC 환경을 유지하는 데 도움이 됩니다.

- **Principle of least privilege:** Always grant users the minimum permissions necessary to perform their duties. Avoid overly broad permissions.
- **Regular review:** Periodically review assigned roles and permissions for users to ensure they are still appropriate and comply with current security policies.
- **Document roles:** Clearly document the purpose and permissions of each custom role you create.
- **Separate duties:** Implement roles that enforce separation of duties to prevent a single user from having too much control over critical processes.

Chapter 23. SSL 인증서

SUSE Multi-Linux Manager는 SSL 인증서를 사용하여 클라이언트가 올바른 서버에 등록되었는지 확인합니다.

SSL을 사용하여 SUSE Multi-Linux Manager 서버에 등록하는 모든 클라이언트는 서버 인증서에 대해 유효성을 검사하여 올바른 서버에 연결하고 있는지 확인합니다. 이 프로세스를 SSL 핸드셰이크라고 합니다.

SSL 핸드셰이크 동안 클라이언트는 서버 인증서의 호스트 이름이 필요한 이름과 일치하는지 확인합니다. 클라이언트는 또한 서버 인증서가 신뢰할 수 있는지 확인해야 합니다.

인증 기관(CA)은 다른 인증서에 서명하기 위해 사용되는 인증서입니다. 모든 인증서는 인증 기관(CA)에서 서명해야 유효한 것으로 간주되며 클라이언트가 인증서에 대해 일치시킬 수 있습니다.

SSL 인증이 올바르게 작동하려면 클라이언트가 루트 CA를 신뢰해야 합니다. 즉, 루트 CA가 모든 클라이언트에 설치되어야 합니다.

SSL 인증의 기본 방법은 SUSE Multi-Linux Manager가 자체 서명 인증서를 사용하는 것입니다. 이 경우 SUSE Multi-Linux Manager가 모든 인증서를 생성했으며 루트 CA가 서버 인증서에 직접 서명합니다.

중간 CA를 사용하는 방법도 있습니다. 이 경우 루트 CA는 중간 CA에 서명합니다. 그런 다음 중간 CA는 원하는 수의 다른 중간 CA에 서명할 수 있으며 마지막 CA는 서버 인증서에 서명합니다. 이를 체인 인증서라고 합니다.

연결된 인증서에서 중간 CA를 사용하는 경우, 루트 CA는 클라이언트에 설치되고 서버 인증서는 서버에 설치됩니다. SSL 핸드셰이크 동안 클라이언트는 루트 CA와 서버 인증서 사이의 전체 중간 인증서 체인을 확인할 수 있어야 하므로, 모든 중간 인증서에 액세스할 수 있어야 합니다.

이를 달성하는 두 가지 주요 방법이 있습니다. 이전 버전의 SUSE Multi-Linux Manager에서는 기본적으로 모든 중간 CA가 클라이언트에 설치됩니다. 그러나 서버에서 서비스를 구성하여 클라이언트에 제공할 수도 있습니다. 이 경우 SSL 핸드셰이크 동안 서버는 서버 인증서와 모든 중간 CA를 제공합니다. 이제 이 방식이 기본 구성으로 사용됩니다.

기본적으로 SUSE Multi-Linux Manager는 중간 CA 없이 자체 서명 인증서를 사용합니다. 추가 보안을 위해 인증서에 서명하도록 타사 CA를 지정할 수 있습니다. 타사 CA는 인증서에 검사를 수행하여 포함된 정보가 올바른지 확인합니다. 그리고 일반적으로 이 서비스에 대해 연간 요금을 청구합니다. 타사 CA를 사용하면 인증서를 스푸핑하기 더 어렵고 설치에 대한 추가 보호가 제공됩니다. 타사 CA에서 서명한 인증서가 있는 경우 SUSE Multi-Linux Manager 설치로 임포트할 수 있습니다.

이 설명서에서는 SSL 인증서 사용을 2단계로 설명합니다.

1. SUSE Multi-Linux Manager 도구를 사용한 자체 서명 인증서 생성 방법
2. SUSE Multi-Linux Manager 서버 또는 프록시에 인증서 배포 방법

자체 또는 외부 PKI와 같은 타사 인스턴스에서 인증서가 제공되는 경우 한 단계를 건너뛸 수 있습니다.

- 자체 서명 인증서 생성에 대한 자세한 내용은 **Administration › Ssl-certs-selfsigned**에서 확인할 수 있습니다.
- 인증서 임포트에 대한 자세한 내용은 **Administration › Ssl-certs-imported**에서 확인할 수 있습니다.

23.1. SUSE Multi-Linux Manager 컨테이너에 SSL 인증서 제공

23.1.1. Podman

SSL 인증서는 podman 시크릿으로 저장되며 각 컨테이너에 할당됩니다. Podman SSL 시크릿은 다음과 같습니다.

- CA 인증서
  - uyuni-ca
  - uyuni-db-ca
- 서버 인증서 및 키
  - uyuni-cert
  - uyuni-key
- 데이터베이스 인증서 및 키
  - uyuni-db-cert
  - uyuni-db-key

23.2. 자체 서명된 SSL 인증서

기본적으로 SUSE Multi-Linux Manager은(는) 자체 서명 인증서를 사용합니다. 이 경우 인증서는 SUSE Multi-Linux Manager에 의해 생성 및 서명됩니다. 이 방법에서는 인증서의 세부 사항이 정확한지 확인하기 위해 독립 인증 기관을 사용하지 않습니다. 타사 CA는 인증서에 포함된 정보가 정확한지 확인하기 위해 검사를 수행합니다.

- 타사 CA에 대한 자세한 내용은 **Administration › Ssl-certs-imported**에서 확인할 수 있습니다.
- 인증서 바꾸기에 대한 자세한 내용은 administration:ssl-certs-imported.pdf에서 확인할 수 있습니다.

이 섹션에서는 신규 또는 기존 설치에서 자체 서명된 인증서를 생성하거나 다시 생성하는 방법을 설명합니다.

SSL 키 및 인증서의 호스트 이름은 이를 배포하는 시스템의 정규화된 호스트 이름과 일치해야 합니다.

23.2.1. 기존 서버 인증서 재생성

기존 인증서가 만료되었거나 어떤 이유로 작동이 중지된 경우 기존 CA에서 새 서버 인증서를 생성할 수 있습니다.

절차: 기존 서버 인증서 재생성

1. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 서버 인증서 재생성:

```
mgrctl exec -ti -- 'rhn-ssl-tool --gen-server --dir="/root/ssl-build" --set
-country="COUNTRY" \
--set-state="STATE" --set-city="CITY" --set-org="ORGANIZATION" \
--set-org-unit="ORGANIZATION UNIT" --set-email="name@example.com" \
--set-hostname="susemanager.example.com" --set-cname="example.com"'
```

set-cname 파라미터가 SUSE Multi-Linux Manager 서버의 정규화된 도메인 이름인지 확인합니다. 여러 별칭이

필요한 경우 **set-cname** 파라미터를 여러 번 사용하면 됩니다.

개인 키 및 서버 인증서는 서버 컨테이너의 **/root/ssl-build/susemanager/** 디렉토리에서 **server.key** 및 **server.crt**로 찾을 수 있습니다. 마지막 디렉토리의 이름은 **--set-hostname** 옵션과 함께 사용된 호스트 이름에 따라 다릅니다.

컨테이너의 호스트 podman 시크릿을 업데이트하여 새 인증서 및 키를 배포하거나 임포트합니다. 방금 생성한 인증서 임포트에 대한 자세한 내용은 administration:ssl-certs-imported.pdf에서 확인할 수 있습니다.

23.2.2. 새 CA 및 서버 인증서 생성

루트 CA를 교체해야 하는 경우에는 유의해야 합니다. 서버와 클라이언트 간의 신뢰 체인이 끊길 수 있습니다. 그러한 경우 관리 사용자가 모든 클라이언트에 로그인하여 CA를 직접 배포해야 합니다.

절차: 새 인증서 생성

1. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 이전 인증서 디렉토리를 새 위치로 이동:

```
mgrctl exec -- mv /root/ssl-build /root/old-ssl-build
```

2. 새 CA 인증서를 생성합니다.

```
mgrctl exec -ti -- 'rhn-ssl-tool --gen-ca --dir="/root/ssl-build" --set
-country="COUNTRY" \
--set-state="STATE" --set-city="CITY" --set-org="ORGANIZATION" \
--set-org-unit="ORGANIZATION UNIT" --set-common-name="SUSE Manager CA Certificate" \
--set-email="name@example.com"'
```

3. 새 서버 인증서 생성:

```
mgrctl exec -ti -- 'rhn-ssl-tool --gen-server --dir="/root/ssl-build" --set
-country="COUNTRY" \
--set-state="STATE" --set-city="CITY" --set-org="ORGANIZATION" \
--set-org-unit="ORGANIZATION UNIT" --set-email="name@example.com" \
--set-hostname="susemanager.example.top" --set-cname="example.com"'
```

set-cname 파라미터가 SUSE Multi-Linux Manager 서버의 정규화된 도메인 이름인지 확인합니다. 여러 별칭이 필요한 경우 **set-cname** 파라미터를 여러 번 사용하면 됩니다.

호스트 이름 및 cname을 사용하여 각 프록시에 대해서도 서버 인증서를 생성해야 합니다.

23.3. SSL 인증서 임포트

이 섹션에서는 새 SUSE Multi-Linux Manager의 설치를 위해 SSL 인증서를 구성하는 방법과 기존 인증서를 교체하는 방법을 설명합니다.

시작하기 전, 다음을 확인해야 합니다.

- 인증 기관(CA) SSL 공개 인증서. CA 체인을 사용하는 경우 모든 중간 CA도 반드시 사용 가능해야 합니다.

- SSL 서버 개인 키
- SSL 서버 인증서
- SSL 데이터베이스 개인 키
- SSL 데이터베이스 인증서

모든 파일은 PEM 형식이어야 합니다.

SSL 서버 인증서의 호스트명은 해당 인증서를 배포하는 머신의 전체 호스트 이름과 일치해야 합니다. 인증서의 **X509v3 Subject Alternative Name** 섹션에서 호스트 이름을 설정할 수 있습니다. 환경에 따라 필요한 경우 여러 호스트 이름을 나열할 수도 있습니다. 지원되는 키 유형은 **RSA** 및 **EC**(Elliptic Curve)입니다.

데이터베이스 SSL 인증서에는 **reportdb** 및 **db**를 **Subject Alternative Name**이 필요합니다.

타사 기관은 일반적으로 중간 CA를 사용하여 요청된 서버 인증서에 서명합니다. 이 경우 체인의 모든 CA를 사용할 수 있어야 합니다. **mgrdadm** 명령이 인증서 정렬을 처리합니다. 루트 CA는 별도의 파일에 위치하는 것이 좋습니다. 서버 인증서 파일은 서버 인증서를 맨 앞에 두고, 그 뒤에 모든 중간 CA 인증서를 순서대로 포함해야 합니다.

23.3.1. 새 설치를 위해 인증서 임포트

By default, SUSE Multi-Linux Manager uses a self-signed certificate. Certificates can be imported with third-party certificates at the installation time.

Procedure: Importing certificates on a new SUSE Multi-Linux Manager server

1. **Installation-and-upgrade › Install-server**의 지침에 따라 SUSE Multi-Linux Manager 서버를 배포합니다. **mgradm install podman**에 올바른 파일을 파라미터로 전달해야 합니다. 파라미터는 다음과 같습니다.

```
타사 SSL 인증서 플래그:
      --ssl-ca-intermediate 문자열   임시 CA 인증서 경로
      --ssl-ca-root 문자열            루트 CA 인증서 경로
      --ssl-server-cert 문자열        서버 인증서 경로
      --ssl-server-key 문자열         서버 키 경로
      --ssl-db-ca-intermediate 문자열         서버와 다른 경우 데이터베이스의 중간
CA 인증서 경로
      --ssl-db-ca-root string          서버와 다른 경우 데이터베이스의 루트 CA
인증서 경로
      --ssl-db-cert string             데이터베이스 인증서 경로
      --ssl-db-key string              데이터베이스 키 경로
```

Intermediate CAs can either be available in the file which is specified with **--ssl-ca-root**, or specified as extra options with **--ssl-ca-intermediate**. The **--ssl-ca-intermediate** option can be specified multiple times.

23.3.2. Import certificates for new proxy installations

The proxy certificates are embedded in the generated configuration. In order to use a third-party certificate, it needs to be provided during the configuration.

Procedure: Importing certificates on a new SUSE Multi-Linux Manager Proxy

1. **Installation-and-upgrade › Install-proxy**의 지침에 따라 SUSE Multi-Linux Manager 프록시를 설치합니다.
2. 프롬프트를 따라 설정을 완료합니다.

Use the same certificate authority (CA) to sign all certificates for servers and proxies. Certificates signed with different CAs do not match.

23.3.3. Replace certificates

You can replace active certificates on your SUSE Multi-Linux Manager installation with a new certificate. There are two cases to consider: replacing only the server or database certificate, and replacing the root CA.

Replacing the root certificate requires more time and planning to avoid disruption as all the registered proxies and systems will need to have the new CA in their database before switching to it at the server level.

When using third-party certificates signed by an intermediate CA, the intermediate CA certificates need to be appended to the server or database certificate file.

The order is important: first comes the server certificate, then the CAs from the one which signed the certificate to the one signed by the root CA. The root CA certificate should not be appended to the server certificate file.

Procedure: Replacing all existing certificates

1. The following considers that you have **root-ca.pem**, **intermediate-ca1.pem**, **intermediate-ca2.pem**, **server.pem** and **server.key** files. It may be different depending on the number of intermediate CAs in the server certificate signature chain.
2. Combine the intermediate CAs and server certificates. The order matters, the server must be first and the intermediate CAs in order. Do not add the root CA last into the chain as it will be passed separately to **uyuni-ca** and **uyuni-db-ca** secrets. If there is no intermediate CA, then you can use the **server.pem** instead of the combined file in the next steps.

```
cat server.pem intermediate-ca1.pem intermediate-ca2.pem >combined-server.pem
```

3. On the SUSE Multi-Linux Manager container host, at the command prompt, recreate podman certificate secrets passing the files paths:

```
podman secret create --replace uyuni-ca $path_to_ca_certificate
podman secret create --replace uyuni-cert $path_to_combined_server_certificate
podman secret create --replace uyuni-key $path_to_server_key

podman secret create --replace uyuni-db-ca $path_to_database_ca_certificate
podman secret create --replace uyuni-db-cert
$path_to_combined_database_certificate
podman secret create --replace uyuni-db-key $path_to_database_key
```

Procedure: Restarting the server

1. 컨테이너 호스트에서 서비스를 재시작하여 변경사항을 적용합니다.

```
mgradm restart
```

프록시를 사용하는 경우, 각 프록시의 호스트 이름과 cname을 사용하여 각 프록시에 대한 서버 인증서 RPM을 생성해야 합니다. 새로운 구성 tarball을 생성하고 배포합니다.

자세한 내용은 installation-and-upgrade:container-deployment/mlm/proxy-deployment-mlm.pdf을 참조하십시오.

If the Root CA was changed, it needs to get deployed to all the clients connected to SUSE Multi-Linux Manager. This is ideally done in advance to minimize the disruption.

CA 인증서가 업데이트된 경우, Kiwi 인증서가 포함된 RPM 파일을 다시 패키징해야 합니다.

SUSE Multi-Linux Manager 서버 컨테이너 호스트에서 다음 명령을 실행합니다.

```
mgrctl exec mgr-package-rpm-certificate-osimage
```

After that, apply highstate on the Image Build hosts to deploy the new certificates for Kiwi to use.

Procedure: Deploying the root CA on clients

1. SUSE Multi-Linux Manager Web UI에서 **Systems › 개요**로 이동합니다.
2. 시스템 설정 관리자에 추가하려면 모든 클라이언트를 선택합니다.
3. **시스템 › 시스템 세트 관리자 › 개요**으로 이동합니다.
4. **상태** 필드에서 **[적용]**을 클릭하여 시스템 상태를 적용합니다.
5. **Highstate** 페이지에서 **[Highstate 적용]**을 클릭하여 변경사항을 클라이언트에 전파합니다.

23.4. HTTP Strict Transport Security

HTTP Strict Transport Security(HSTS)는 프로토콜 다운그레이드 공격 및 쿠키 하이재킹과 같은 가로채기 공격으로부터 웹 사이트를 보호하는 데 도움이 되는 정책 메커니즘입니다.

SUSE Multi-Linux Manager에서는 HSTS가 기본적으로 활성화되어 있습니다. 서버에서 HSTS를 비활성화해야 하는 경우 다음 절차를 따르십시오.

절차: 서버에서 HSTS 비활성화

1. 서버 컨테이너 호스트에서 루트 권한으로 다음 명령을 실행하여 **max-age=0** 설정이 포함된 새 구성 파일을 생성합니다.

```
mgrctl exec -- \
  echo 'Header always set Strict-Transport-Security "max-age=0;
includeSubDomains"' \
  > /etc/apache2/conf.d/zz-spacewalk-www-hsts.conf
```

2. 다음 명령으로 Apache를 재시작합니다.

```
mgrctl exec -- systemctl restart apache2
```

프록시에서 이를 비활성화해야 하는 경우 다음 절차를 따르십시오.

절차: 프록시에서 HSTS 비활성화

1. 서버 컨테이너 호스트에서 루트 권한으로 다음 명령을 실행하여 **max-age=0** 설정이 포함된 새 구성 파일을 생성합니다.

```
echo 'Header always set Strict-Transport-Security "max-age=0;
includeSubDomains' \
  > /etc/uyuni/custom-httpd.conf
```

2. 다음 명령 실행:

```
mgrpxy install podman --tuning-httpd /etc/uyuni/custom-httpd.conf config.tar.gz
```

새 설정 파일의 이름을 **<filename>.conf**로 지정하는 경우 적절한 타이밍에 로드되어야 합니다. 예를 들어, **spacewalk-www.conf**에 정의된 항목을 재정의하려면 이 파일 뒤에 새 파일을 알파벳순으로 붙여야 합니다. Apache가 파일을 로드하는 방법에 대한 자세한 내용은 https://httpd.apache.org/docs에서 확인할 수 있습니다.

SUSE Multi-Linux Manager에서 생성한 기본 SSL 인증서 또는 자체 서명된 인증서를 사용하는 동안 HSTS가 활성화되면 해당 인증서에 서명하기 위해 사용된 CA가 브라우저에서 신뢰하는 경우를 제외하고 브라우저는 HTTPS를 통한 연결을 거부합니다. SUSE Multi-Linux Manager에서 생성한 SSL 인증서를 사용하는 경우, http://<SERVER-HOSTNAME>/pub/RHN-ORG-TRUSTED-SSL-CERT에 있는 파일을 모든 사용자의 브라우저로 임포트하여

신뢰할 수 있습니다.

Chapter 24. 구독 일치

SUSE 제품에는 SUSE Customer Center(SCC)에서 관리하는 구독이 필요합니다. SUSE Multi-Linux Manager는 야간 보고서를 실행하여 SCC 계정에 등록된 모든 클라이언트의 구독 상태를 확인합니다. 보고서는 어떤 클라이언트가 어떤 구독을 이용하는지, 얼마나 남아 있고 사용할 수 있는지, 현재 구독이 없는 클라이언트에 대한 정보를 제공합니다.

보고서를 살펴보려면 **감사 › 구독 일치**로 이동합니다.

구독 보고서 탭에서는 현재 및 만료 예정 구독에 대한 정보를 확인할 수 있습니다.

일치하지 않는 제품 보고서 탭에서는 현재 구독이 없는 클라이언트 목록을 확인할 수 있습니다. 여기에는 일치하지 않거나 현재 SUSE Multi-Linux Manager에 등록되지 않은 클라이언트가 포함됩니다. 보고서에는 제품 이름과 일치하지 않는 시스템 수가 포함됩니다.

핀 탭을 사용하면 개별 클라이언트를 관련 구독에 연결할 수 있습니다. 이는 구독 관리자가 클라이언트를 구독에 자동으로 연결하지 않는 경우에 특히 유용합니다.

메시지 탭에는 일치 프로세스에서 구독 매처가 생성한 모든 메시지가 표시됩니다. 이러한 메시지는 결과를 이해하고 일치를 개선하는 데 도움이 되는 정보를 제공합니다.

보고서를 .csv 형식으로 다운로드하거나 **/var/lib/spacewalk/subscription-matcher/** 디렉토리의 해당 명령 프롬프트에서 보고서에 액세스할 수도 있습니다.

기본적으로 구독 선택기는 매일 자정에 실행됩니다. 이를 변경하려면 **관리자 › 작업 스케줄**으로 이동하여 **gatherer-matcher-default**를 클릭합니다. 필요에 따라 일정을 변경하고 **[일정 업데이트]**를 클릭합니다.

보고서는 현재 구독이 있는 현재 클라이언트만 일치시킬 수 있기 때문에, 시간이 지남에 따라 일치 항목이 변경됩니다. 동일한 클라이언트가 항상 동일한 구독과 일치하는 것은 아닙니다. 이는 새 클라이언트가 등록 또는 등록 취소되었거나 구독이 추가 또는 만료되었기 때문일 수 있습니다.

구독 선택기는 계정의 구독 약관에 따라 일치하지 않는 제품의 수를 자동으로 줄이려고 시도합니다. 그러나 불완전한 하드웨어 정보, 알 수 없는 가상 머신 호스트 할당 또는 알 수 없는 공용 클라우드에서 실행 중인 클라이언트가 있는 경우 선택기는 사용할 수 있는 구독이 충분하지 않음을 표시할 수 있습니다. 정확성을 보장하기 위해, 항상 SUSE Multi-Linux Manager에 포함된 고객 관련 데이터가 완전한지 확인하십시오.

구독 선택기가 항상 클라이언트와 구독을 정확하게 일치시키는 것은 아니며, 감사를 대체하기 위한 용도가 아닙니다.

24.1. 구독에 클라이언트 고정

구독 선택기가 특정 클라이언트를 올바른 구독과 자동으로 일치시키지 않는 경우 수동으로 고정할 수 있습니다. 핀을 생성하면 구독 선택기는 특정 구독을 주어진 시스템 또는 시스템 그룹과 일치시키는 것을 선호합니다.

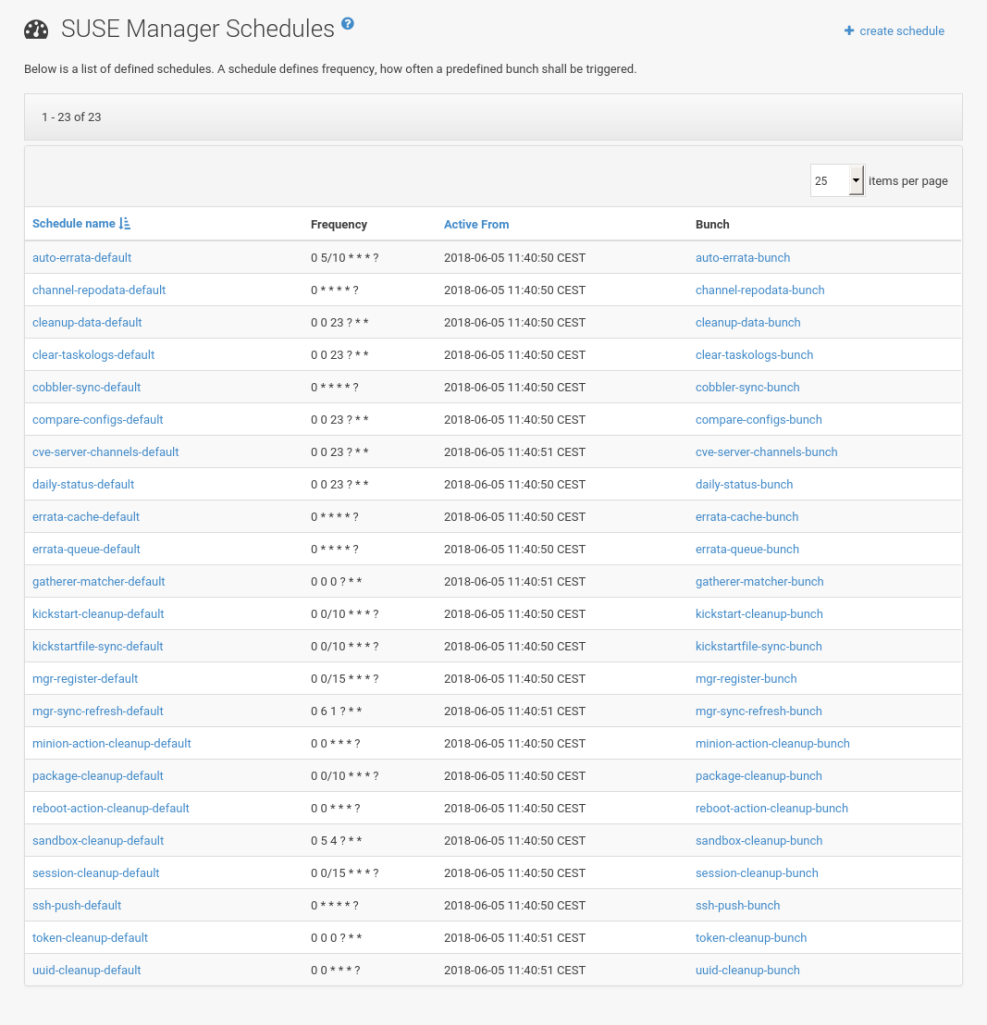
그러나 선택기가 항상 핀을 따르는 것은 아닙니다. 사용할 수 있는 구독 및 구독을 클라이언트에 적용할 수 있는지 여부에 따라 다릅니다. 또한, 구독의 이용 약관을 위반하는 일치를 초래하는 경우 또는 핀이 무시되는 경우 일치자가 더 정확한 일치를 감지하면 핀이 무시됩니다.

새 핀을 추가하려면 **[핀 추가]**를 클릭하고 고정할 클라이언트를 선택합니다.

고정을 정기적으로 사용하거나 많은 수의 클라이언트에 대해서는 사용하지 않는 것이 좋습니다. 구독 선택기 도구는 일반적으로 대부분의 설치에서 충분히 정확합니다.

Chapter 25. 작업 스케줄

사전 정의된 모든 작업 묶음은 **관리자 › 작업 일정** 아래에 나열됩니다.



Schedule name	Frequency	Active From	Bunch
auto-errata-default	0 5/10 * * * ?	2018-06-05 11:40:50 CEST	auto-errata-bunch
channel-repodata-default	0 * * * * ?	2018-06-05 11:40:50 CEST	channel-repodata-bunch
cleanup-data-default	0 0 23 ? * *	2018-06-05 11:40:50 CEST	cleanup-data-bunch
clear-taskologs-default	0 0 23 ? * *	2018-06-05 11:40:50 CEST	clear-taskologs-bunch
cobbler-sync-default	0 * * * * ?	2018-06-05 11:40:50 CEST	cobbler-sync-bunch
compare-configs-default	0 0 23 ? * *	2018-06-05 11:40:50 CEST	compare-configs-bunch
cve-server-channels-default	0 0 23 ? * *	2018-06-05 11:40:51 CEST	cve-server-channels-bunch
daily-status-default	0 0 23 ? * *	2018-06-05 11:40:50 CEST	daily-status-bunch
errata-cache-default	0 * * * * ?	2018-06-05 11:40:50 CEST	errata-cache-bunch
errata-queue-default	0 * * * * ?	2018-06-05 11:40:50 CEST	errata-queue-bunch
gatherer-matcher-default	0 0 0 ? * *	2018-06-05 11:40:51 CEST	gatherer-matcher-bunch
kickstart-cleanup-default	0 0/10 * * * ?	2018-06-05 11:40:50 CEST	kickstart-cleanup-bunch
kickstartfile-sync-default	0 0/10 * * * ?	2018-06-05 11:40:50 CEST	kickstartfile-sync-bunch
mgr-register-default	0 0/15 * * * ?	2018-06-05 11:40:50 CEST	mgr-register-bunch
mgr-sync-refresh-default	0 6 1 ? * *	2018-06-05 11:40:51 CEST	mgr-sync-refresh-bunch
minion-action-cleanup-default	0 0 * * * ?	2018-06-05 11:40:50 CEST	minion-action-cleanup-bunch
package-cleanup-default	0 0/10 * * * ?	2018-06-05 11:40:50 CEST	package-cleanup-bunch
reboot-action-cleanup-default	0 0 * * * ?	2018-06-05 11:40:50 CEST	reboot-action-cleanup-bunch
sandbox-cleanup-default	0 5 4 ? * *	2018-06-05 11:40:50 CEST	sandbox-cleanup-bunch
session-cleanup-default	0 0/15 * * * ?	2018-06-05 11:40:50 CEST	session-cleanup-bunch
ssh-push-default	0 * * * * ?	2018-06-05 11:40:50 CEST	ssh-push-bunch
token-cleanup-default	0 0 0 ? * *	2018-06-05 11:40:51 CEST	token-cleanup-bunch
uuid-cleanup-default	0 0 * * * ?	2018-06-05 11:40:51 CEST	uuid-cleanup-bunch

SUSE Multi-Linux Manager 일정 › 일정 이름을 클릭하면 **일정 이름 › 기본 일정 세부사항**이 열립니다. 비활성화하거나 빈도를 변경할 수 있습니다.

설정과 함께 일정을 업데이트하려면 **[일정 편집]**을 클릭하십시오.

일정을 비활성화하려면 오른쪽 상단의 **[일정 비활성화]**를 클릭합니다.

일정은 SUSE Multi-Linux Manager의 올바른 작동을 위해 필수이므로 필요하다고 확신하는 경우에만 일정을 비활성화해야 합니다.

작업이 비활성화되어도 목록에는 계속 표시됩니다. **SUSE Multi-Linux Manager 일정 › 일정 이름**을 클릭하면 **[일정 활성화]**를 눌러 작업을 다시 활성화할 수 있습니다.

무리 이름을 클릭하면 해당 묶음 유형의 실행 목록 및 해당 상태가 표시됩니다.

시작 시간 링크를 클릭하면 **일정 이름 › 기본 일정 세부 사항**으로 돌아갑니다.

25.1. 사전 정의된 작업 묶음

다음과 같은 사전 정의 작업 묶음은 기본적으로 예약되며 구성할 수 있습니다.

errata-cache-default:

필요한 경우 자동 정오표 업데이트를 예약합니다.

channel-repodata-default:

리포지토리 메타데이터 파일을 (다시) 생성합니다.

cleanup-data-default:

데이터베이스에서 부실한 패키지 변경 로그 및 모니터링 시계열 데이터를 정리합니다.

clear-taskologs-default:

데이터베이스에서 작업 유형에 따라 지정된 일 수를 초과한 작업 엔진(taskomatic) 과거 기록 데이터를 지웁니다.

cobbler-sync-default:

SUSE Multi-Linux Manager에서 Cobbler로 배포 및 프로파일 데이터를 동기화합니다. Cobbler에서 제공하는 자동 설치에 대한 자세한 내용은 **Client-configuration › Autoinst-intro**에서 확인할 수 있습니다.

compare-configs-default:

구성 채널에 저장된 구성 파일과 모든 구성 가능 서버에 저장된 파일을 비교합니다. 비교를 검토하려면 **시스템** 탭을 클릭하고 관심이 있는 시스템을 선택합니다. **구성 › 파일 비교**로 이동합니다. 자세한 내용은 reference:systems/system-details/sd-configuration.pdf에서 확인할 수 있습니다.

cve-server-channels-default:

감사 › CVE 감사 페이지에 결과를 표시하기 위해 사용되는 미리 계산된 내부 CVE 데이터를 업데이트합니다. **감사 › CVE 감사** 페이지의 검색 결과는 이 일정의 마지막 실행으로 업데이트됩니다. 자세한 내용은 **Reference › Audit**에서 확인할 수 있습니다.

daily-status-default:

관련 주소로 일일 보고서 이메일을 전송합니다. 특정 사용자에 대한 알림을 구성하는 방법에 대한 자세한 내용은 **Reference › Users**에서 확인할 수 있습니다.

errata-advisory-map-sync-default

내부 SUSE 패치 공급업체 권고 사항 데이터베이스 테이블을 업데이트합니다. 가능한 경우, SUSE에서 제공한 원본 권고 사항은 각 패치 상세 정보의 공급업체 권고 사항 섹션에 표시됩니다.

errata-cache-default:

내부 패치 캐시 데이터베이스 테이블을 업데이트하며, 이는 각 서버에 대한 업데이트가 필요한 패키지를 찾기 위한 용도로 사용됩니다. 또한, 특정 패치에 관심이 있을 수 있는 사용자에게 알림 이메일도 전송합니다. 패치에 대한 자세한 내용은 **Reference › Patches**에서 확인할 수 있습니다.

errata-queue-default:

자동 업데이트(패치)를 수신하도록 구성된 서버를 대기열에 추가합니다.

gatherer-matcher-default

Virtual Host Managers에 구성된 Virtual Host Gatherer를 실행하여 가상 호스트 데이터를 수집합니다. 업데이트된 데이터를 사용할 수 있게 되면 Subscription Matcher 작업이 실행됩니다.

kickstart-cleanup-default:

오래된 Kickstart 세션 데이터를 정리합니다.

kickstartfile-sync-default

구성 마법사에서 생성한 Kickstart 프로파일에 해당하는 Cobbler 파일을 생성합니다.

mgr-forward-registration-default

클라이언트 등록 데이터를 SUSE Customer Center과(와) 동기화합니다. 기본적으로 신규, 변경 또는 삭제된 클라이언트 데이터가 전달됩니다. **/etc/rhn/rhn.conf**에 설정된 동기화를 비활성화하려면 다음을 수행합니다.

```
server.susemanager.forward_registration = 0
```

SCC와의 데이터 동기화를 비활성화하면 RMT, SMT, SUSE Multi-Linux Manager 및 SCC에 직접 등록된 클라이언트 사이에서 관리형 클라이언트의 가시성이 저하됩니다.

데이터를 동기화하면 등록된 모든 클라이언트를 일관되게 확인할 수 있습니다.

서비스 개선에 도움이 되도록 거부하시는 이유를 알려주십시오.

mgr-sync-refresh-default

SUSE Customer Center과(와)동기화(**mgr-sync-refresh**)합니다. 기본적으로 모든 사용자 정의 채널도 이 작업의 일부로 동기화됩니다. 사용자 정의 채널 동기화에 대한 자세한 내용은 administration:custom-channels.pdf에서 확인할 수 있습니다.

minion-action-chain-cleanup-default

오래된 작업 체인 데이터를 정리합니다.

minion-action-cleanup-default

파일 시스템에서 부실한 클라이언트 작업 데이터를 삭제합니다. 우선, Salt 작업 캐시에 저장된 해당 결과를 검색하여 완료되지 않은 작업의 완료를 시도합니다. 서버에서 작업 결과가 누락된 경우, 완료되지 않은 작업이 발생할 수 있습니다. 성공적으로 완료된 작업의 경우 실행된 스크립트 파일 등의 아티팩트를 제거합니다.

minion-checkin-default

클라이언트에 대한 정기 체크인을 수행합니다.

notifications-cleanup-default

만료된 알림 메시지를 정리합니다.

oval-data-sync-default

CVE 감사 쿼리의 정확도를 향상하기 위해 필수 OVAL 데이터를 생성합니다.

package-cleanup-default

파일 시스템에서 부실 패키지 파일을 삭제합니다.

reboot-action-cleanup-default

여섯 시간을 초과하여 대기 중인 모든 재부팅 작업은 실패한 것으로 표시되고 관련 데이터가 데이터베이스에서 정리됩니다. 재부팅 작업 예약에 대한 자세한 내용은 reference:systems/system-details/sd-provisioning.pdf에서 확인할 수 있습니다.

session-cleanup-default

부실 웹 인터페이스 세션, 일반적으로 사용자가 로그인할 때 임시로 저장한 후 로그아웃하기 전에 브라우저를 닫는 데이터를 정리합니다.

ssh-service-default

클라이언트가 **SSH 푸시** 연락 방법으로 구성된 경우 SSH를 통해 SUSE Multi-Linux Manager(으)로 체크인하라는 메시지가 표시됩니다. 또한 재부팅 후 작업 체인을 재개합니다.

system-overview-update-queue-default

시스템 개요 데이터를 업데이트합니다.

system-profile-refresh-default

모든 시스템에서 하드웨어 새로 고침을 실행합니다. 이 작업은 매월 수행되며 SUSE Multi-Linux Manager 서버의 부하를 증가시킬 수 있습니다. 이 작업에서는 **Specialized-guides › Salt**을 사용합니다. 배치 크기 조정에 대한 내용은 specialized-guides:large-deployments/tuning.pdf에서 확인할 수 있습니다.

token-cleanup-default

Salt 클라이언트가 패키지 및 메타데이터를 다운로드하기 위해 사용하는 만료된 리포지토리 토큰을 삭제합니다.

update-payg-default

구성된 PAYG 클라우드 인스턴스에서 인증 데이터를 수집합니다.

update-reporting-default

로컬 보고 데이터베이스를 업데이트합니다.

update-reporting-hub-default

주변 SUSE Multi-Linux Manager 서버에서 모든 보고 데이터를 수집하고 Hub Reporting Database를 업데이트합니다.

update-system-overview-default

시스템 개요 데이터가 최신 상태인지 정기적으로 확인합니다.

uuid-cleanup-default

오래된 UUID 레코드를 정리합니다.

Chapter 26. 변경 로그 튜닝

일부 패키지의 경우 변경 로그 항목 목록이 깁니다. 이 데이터는 기본적으로 다운로드되지만, 유지하는 것이 항상 유용한 정보는 아닙니다. 다운로드되는 변경 로그 메타데이터의 양을 제한하고 디스크 공간을 절약하기 위해 디스크에 유지할 항목 수를 제한할 수 있습니다.

서버 컨테이너 내에서 단계를 실행하기 전에 **mgrctl term**을 사용합니다.

이 구성 옵션은 **/etc/rhn/rhn.conf** 구성 파일에 있습니다. 파라미터의 기본값은 **20**입니다. 이 값을 **0**으로 변경하면 항목 수가 무제한으로 표시됩니다.

```
java.max_changelog_entries = 20
```

이 파라미터를 설정하면 새 패키지가 동기화될 때에만 적용됩니다.

이 파라미터를 변경한 후 **mgradm restart**로 서비스를 재시작합니다.

캐시된 데이터를 삭제하고 재생성하여 오래된 데이터를 제거할 수 있습니다.

캐시된 데이터를 삭제 및 재생성하기 위해서는 시간이 오래 걸릴 수 있습니다. 보유하고 있는 채널 수와 삭제할 데이터의 양에 따라 몇 시간이 걸릴 수 있습니다. 작업은 Taskomatic에 의해 백그라운드에서 실행되므로 작업이 완료되는 동안 SUSE Multi-Linux Manager을 계속 사용할 수 있지만 성능이 약간 손실될 수 있습니다.

명령줄에서 캐시된 데이터를 삭제하고 재생성을 요청합니다.

```
spacewalk-sql -i
```

그런 다음 SQL 데이터베이스 프롬프트에 다음을 입력합니다.

```
DELETE FROM rhnPackageRepodata;
INSERT INTO rhnRepoRegenQueue (id, CHANNEL_LABEL, REASON, FORCE)
(SELECT sequence_nextval('rhn_repo_regen_queue_id_seq'),
        C.label,
        'cached data regeneration',
        'Y'
        FROM rhnChannel C);
\q
```

Chapter 27. 사용자

SUSE Multi-Linux Manager 관리자는 새 사용자를 추가하고 권한을 부여하며 사용자를 비활성화 또는 삭제할 수 있습니다. 대규모 사용자를 관리하는 경우 사용자를 시스템 그룹에 할당하여 그룹 수준에서 권한을 관리할 수 있습니다. 언어 및 테마 기본값을 포함하여 Web UI의 시스템 기본값을 변경할 수도 있습니다.

사용자 메뉴는 SUSE Multi-Linux Manager 관리자 계정으로 로그인한 경우에만 사용할 수 있습니다.

SUSE Multi-Linux Manager 사용자를 관리하려면 **사용자 › 사용자 목록 › 전체**로 이동하여 SUSE Multi-Linux Manager 서버의 모든 사용자를 확인합니다. 목록의 각 사용자와 관련해서는 사용자 이름, 실명, 할당된 역할, 사용자가 마지막으로 로그인한 날짜 및 사용자의 현재 상태가 표시됩니다. **[사용자 생성]**을 클릭하여 새 사용자 계정을 생성합니다. 사용자 이름을 클릭하여 **사용자 세부 사항** 페이지로 이동합니다.

조직에 새 사용자를 추가하려면 **[사용자 생성]**을 클릭하고 새 사용자에 대한 세부 사항을 입력한 후 **[로그인 생성]**을 클릭합니다.

27.1. 비밀번호 요구사항

SUSE Multi-Linux Manager은(는) 기본값으로 선택되어 배송됩니다.

모든 새 사용자 비밀번호가 조직의 보안 표준을 준수할 수 있도록 SUSE Multi-Linux Manager 관리자에는 비밀번호 생성 규칙을 강제 적용할 수 있는 옵션이 있습니다.

Web UI에서 **관리자 › 관리자 구성 › 비밀번호 정책**으로 이동하여 비밀번호 요구 사항을 정의합니다. 다음 필드를 조합하여 사용하십시오.

비밀번호 최소 길이

이 필드를 사용하여 비밀번호의 최소 길이를 정의합니다.

비밀번호 최대 길이

이 필드를 사용하여 비밀번호의 최대 길이를 정의합니다.

숫자 필요

이 필드를 사용하여 비밀번호에 숫자(0~9)를 포함해야 하는지 여부를 지정합니다.

소문자 필요

이 필드를 사용하여 비밀번호에 소문자(a~z)를 포함해야 하는지 여부를 지정합니다.

대문자 필요

이 필드를 사용하여 비밀번호에 대문자(A~Z)를 포함해야 하는지 여부를 지정합니다.

연속 문자 제한

이 필드를 사용하여 연속 문자를 제한할지 여부를 지정합니다.

특수 문자 필요

이 필드를 사용하여 비밀번호에 특수 문자를 포함해야 하는지 여부를 지정합니다.

허용되는 특수 문자

이 필드는 **특수 문자 필요**가 선택된 경우에만 활성화됩니다. 이 필드를 사용하여 허용되는 특수 문자(예: **!@#$%&***)를 지정하십시오.

문자 발생 횟수 제한

이 필드를 사용하여 제한된 문자 발생 횟수를 지정합니다.

최대 문자 발생 횟수

이 필드를 사용하여 최대 문자 발생 횟수를 지정합니다.

[저장] 버튼을 클릭하여 변경된 비밀번호 설정을 저장합니다.

[초기화]를 사용하여 설정을 기본값으로 변경합니다.

SUSE Multi-Linux Manager은(는) 다음 기본값으로 배송됩니다.

- 최소 비밀번호 길이: 4
- 최대 비밀번호 길이: 32
- 대문자 필요: 선택됨

27.2. 계정 비활성화 및 삭제

더 이상 필요하지 않은 사용자 계정을 비활성화 또는 삭제할 수 있습니다. 비활성화된 사용자 계정은 언제든지 다시 활성화할 수 있습니다. 삭제된 사용자 계정은 표시되지 않으며 검색할 수 없습니다.

사용자는 자신의 계정을 비활성화할 수 있습니다. 그러나 사용자에게 관리자 역할이 있는 경우 해당 역할을 제거해야 계정을 비활성화할 수 있습니다.

비활성화된 사용자는 SUSE Multi-Linux Manager Web UI에 로그인하거나 작업을 예약할 수 없습니다. 비활성화되기 전에 사용자가 예약한 작업은 작업 대기열에 남아 있습니다. 비활성화된 사용자는 SUSE Multi-Linux Manager 관리자가 다시 활성화할 수 있습니다.

27.3. 사용자 역할

사용자에게는 여러 역할을 할당할 수 있으며, 1개의 역할에 언제든지 2명 이상의 사용자가 할당될 수 있습니다. 항상 1명 이상의 활성 SUSE Multi-Linux Manager 관리자가 있어야 합니다.

SUSE Multi-Linux Manager 관리자 역할을 제외한 사용자의 역할을 변경하려면, **사용자 › 사용자 목록 › 전체**로 이동하여 변경할 사용자를 선택하고 필요에 따라 관리자 역할을 선택하거나 선택을 취소합니다.

사용자의 SUSE Multi-Linux Manager 관리자 역할을 변경하려면 **관리자 › 사용자**로 이동하여 필요에 따라 **SUSE Multi-Linux Manager 관리자이십니까?**를 선택하거나 선택을 취소합니다.

표 15. 사용자 역할 권한

역할 이름	설명
SUSE Multi-Linux Manager 관리자	다른 사용자의 권한 변경 등 모든 기능을 수행할 수 있습니다.
조직 관리자	활성화 키, 구성, 채널 및 시스템 그룹을 관리합니다.
활성화 키 관리자	활성화 키를 관리합니다.
이미지 관리자	이미지 프로파일, 빌드 및 저장을 관리합니다.
구성 관리자	시스템 구성을 관리합니다.
채널 관리자	채널을 전 세계적으로 구독할 수 있도록 하고 새 채널 생성 등 소프트웨어 채널을 관리합니다.
시스템 그룹 관리자	시스템 그룹 생성 및 삭제, 기존 그룹에 클라이언트 추가, 그룹에 대한 사용자 액세스 관리 등 시스템 그룹을 관리합니다.
일반 사용자	표준 수준의 액세스 권한을 제공합니다. 새로 생성된 사용자는 이 역할에 자동으로 할당됩니다.

27.4. 추가 역할 만들기

SUSE Multi-Linux Manager의 역할 기반 접근 제어(RBAC)를 사용하면 추가 역할을 생성해 사용자 권한을 세밀하게 조정할 수 있습니다. 역할 관리 방법에 대한 자세한 내용은 **Administration › Role-based-access-control**에서 확인할 수 있습니다.

27.5. 사용자 권한 및 시스템

클라이언트를 관리하기 위해 시스템 그룹을 생성한 경우 관리할 사용자에게 그룹을 할당할 수 있습니다.

사용자를 시스템 그룹에 할당하려면 **사용자 › 사용자 목록**으로 이동하여 편집할 사용자 이름을 클릭하고 **시스템 그룹** 탭으로 이동합니다. 할당할 그룹을 확인하고 **[기본값 업데이트]**를 클릭합니다.

사용자에 기본 시스템 그룹을 한 개 이상 선택할 수도 있습니다. 사용자가 새 클라이언트를 등록하면 기본적으로 선택한 시스템 그룹에 할당됩니다. 그러면 사용자는 새로 등록된 클라이언트에 즉시 액세스할 수 있습니다.

외부 그룹을 관리하려면 **사용자 › 시스템 그룹 구성**으로 이동하고 **외부 인증** 탭으로 이동합니다. **[외부 그룹 생성]**을 클릭하여 새 외부 그룹을 생성합니다. 그룹에 이름을 할당하고 적절한 시스템 그룹에 할당합니다.

시스템 그룹에 대한 자세한 내용은 **Reference › Systems**에서 확인할 수 있습니다.

사용자가 관리할 수 있는 개별 클라이언트를 확인하려면 **사용자 › 사용자 목록**으로 이동하여 편집할 사용자 이름을 클릭하고 **시스템** 탭으로 이동합니다. 일괄 작업을 수행하려면 목록에서 클라이언트를 선택하여 시스템 세트 관리자에 추가할 수 있습니다.

시스템 세트 관리자에 대한 자세한 내용은 **Client-configuration › System-set-manager**에서 확인할 수 있습니다.

27.6. 사용자 및 채널 권한

채널의 컨텐트를 사용하는 구독자 또는 채널을 직접 관리할 수 있는 관리자로 조직 내의 소프트웨어 채널에 사용자를 할당할 수 있습니다.

사용자가 채널을 구독하려면, **사용자 › 사용자 목록**으로 이동하여 편집할 사용자 이름을 클릭하고 **채널 권한 › 구독** 탭으로 이동합니다. 할당할 채널을 확인하고 **[권한 업데이트]**를 클릭합니다.

사용자에게 채널 관리 권한을 부여하려면 **사용자 › 사용자 목록**으로 이동하여 편집할 사용자 이름을 클릭하고 **채널 권한 › 관리** 탭으로 이동합니다. 할당할 채널을 확인하고 **[권한 업데이트]**를 클릭합니다.

목록의 일부 채널은 구독할 수 없습니다. 왜냐하면 일반적으로 사용자 관리자 상태 또는 채널 전역 설정 때문입니다.

27.7. 사용자 기본 언어

새 사용자를 생성할 때 Web UI에 사용할 언어를 선택할 수 있습니다. 사용자를 생성한 후 **홈 › 내 기본 설정**으로 이동하여 언어를 변경할 수 있습니다.

기본 언어는 **rhn.conf** 구성 파일에 구성되어 있습니다. 기본 언어를 변경하려면 **/etc/rhn/rhn.conf** 파일을 열고 다음 라인을 추가하거나 편집합니다.

```
web.locale = <LANGCODE>
```

파라미터가 설정되지 않은 경우 기본 언어는 **en_US**입니다.

다음은 SUSE Multi-Linux Manager에서 사용할 수 있는 언어입니다.

표 16. 사용할 수 있는 언어 코드

언어 코드	언어	지역
en_US	영어	미국
zh_CN	중국어	본토, 간체

27.7.1. 사용자 기본 인터페이스 테마

기본적으로 SUSE Multi-Linux Manager Web UI는 사용자가 설치한 제품에 적합한 테마를 사용합니다. Uyuni 또는 SUSE Multi-Linux Manager 색상을 반영하도록 테마를 변경할 수 있습니다. SUSE Multi-Linux Manager 테마에는 어두운 옵션도 있습니다.

rhn.conf 구성 파일에서 기본 테마를 변경할 수 있습니다. 기본 테마를 변경하려면 **/etc/rhn/rhn.conf** 파일을 열고 다음 라인을 추가하거나 편집합니다.

```
web.theme_default = <THEME>
```

표 17. 사용할 수 있는 웹 UI 테마

테마 이름	색상	스타일
suse-light	SUSE Multi-Linux Manager	밝게
suse-dark	SUSE Multi-Linux Manager	어둡게
uyuni	Uyuni	밝게

Chapter 28. 지원

SUSE Multi-Linux Manager에서 SUSE의 지원을 받을 수 있는 권한이 있는 시스템을 관리할 때, **supportconfig** 또는 **sosreport** 같은 지원 데이터를 얻을 수 있습니다. 사용자는 관리형 클라이언트에서 데이터를 수집하여 SUSE 글로벌 기술 지원에 직접 업로드할 수 있습니다.

28.1. 서비스 요청 번호 만들기

글로벌 기술 지원에 지원 데이터를 전달하기 전에 먼저 서비스 요청 번호를 생성해야 합니다.

서비스 요청을 생성하려면 https://scc.suse.com/support/requests로 이동하여 화면의 안내를 따르십시오. 서비스 요청 번호를 기록해 두십시오.

Privacy statement

SUSE는 시스템 보고서를 기밀 데이터로 취급합니다. SUSE의 개인정보 보호 약속에 대한 자세한 내용은 https://www.suse.com/company/policies/privacy/에서 확인할 수 있습니다.

28.2. SUSE Multi-Linux Manager에서 SUSE로 지원 데이터를 수집하고 업로드합니다.

절차: SUSE Multi-Linux Manager Web UI를 사용한 지원 데이터 수집 및 업로드

1. SUSE Multi-Linux Manager Web UI에서 **시스템 › 시스템 목록 › 모두**으로 이동하고 해당 케이스에 추가해야 할 지원 데이터가 있는 클라이언트를 선택합니다.
2. 그런 다음 **세부 사항** 탭으로 이동하여 **지원** 하위 탭을 선택합니다.
3. **Support Case Number** 필드에 위에서 생성한 서비스 요청 번호를 입력합니다.
4. **Upload Region** 필드에서 데이터를 업로드할 서버에 따라 **EU** 또는 **US** 옵션을 선택합니다.
5. **Command-line Arguments** 필드에는 대상 시스템에서 데이터 수집을 위해 사용되는 도구에 대한 옵션을 입력할 수 있습니다. 사용될 도구는 이 필드 아래의 팁에서 확인할 수 있습니다.
6. **시작 시간** 필드에서 이 작업을 실행할 시간을 선택합니다.
7. **[일정]** 버튼을 클릭합니다. 작업이 예약되어 정해진 시간에 실행됩니다. 클라이언트에서 데이터를 수집하여 업로드 서버에 직접 업로드합니다.

지원 데이터를 수집할 수 있는 지원되는 제품:

- SUSE Multi-Linux Manager 프록시

- 허브 시나리오에서 주변기기 서버로 등록된 SUSE Multi-Linux Manager 서버
- 모든 SUSE Linux Enterprise 및 openSUSE 클라이언트
- SUSE Liberty 및 호환 클라이언트
- Ubuntu 클라이언트
- Debian 클라이언트

기본 SUSE Multi-Linux Manager 서버의 지원 데이터를 업로드하려면 컨테이너 호스트에서 **mgradm support config**를 사용합니다.

Chapter 29. 문제 해결

이 섹션에는 SUSE Multi-Linux Manager에서 발생할 수 있는 몇 가지 일반적인 문제와 해결 방법이 포함되어 있습니다.

공용 클라우드와 관련된 문제 해결 주제는 별도로 설명이 제공됩니다.

공용 클라우드 문제 해결에 대한 내용은 **Specialized-guides › Public-cloud-guide**에서 확인할 수 있습니다.

29.1. 자동 설치 문제 해결

통신을 위해 일반 Salt(salt-minion) 구현을 사용하는 경우에는 구성된 클라이언트 채널에서 필요한 모든 소프트웨어(소프트웨어 패키지)를 올바르게 사용할 수 있는지 직접 확인해야 합니다. 이 구현을 SUSE Multi-Linux Manager과(와) 함께 사용하는 것은 권장되지 않습니다.

기본 Salt 번들(venv-salt-minion)을 구현하면 클라이언트 기본 채널과 호환되는 하위 채널로 클라이언트 도구 채널만 있으면 됩니다. 모든 필수 패키지는 Salt 번들의 일부가 됩니다.

- 자동 설치 프로파일의 기본 채널과 관련이 있는 클라이언트 도구 소프트웨어 채널이 조직 및 사용자에게 제공되는지 확인합니다.
- 도구 채널이 SUSE Multi-Linux Manager에 하위 채널로 제공되는지 확인합니다.
- 일반 Salt(salt-minion) 구현을 사용하는 경우에만 관련 채널에서 필수 패키지와 모든 종속성을 사용할 수 있는지도 확인합니다.

29.2. 수명 종료 제품을 위한 리포지토리 부트스트랩 문제 해결

지원하는 제품이 동기화되면 부트스트랩 리포지토리가 자동으로 생성되고 SUSE Multi-Linux Manager 서버에서 재생성됩니다. 제품이 수명 종료에 도달하여 지원이 중단된 경우 제품을 계속 사용하고 싶으면 부트스트랩 리포지토리를 수동으로 생성해야 합니다.

부트스트랩 리포지토리에 대한 자세한 내용은 **Client-configuration › Bootstrap-repository**를 참조하십시오.

절차: 수명 종료 제품의 부트스트랩 리포지토리 생성

1. SUSE Multi-Linux Manager 컨테이너 호스트의 명령 프롬프트에서 루트 권한으로 서버 컨테이너를 입력:

```
mgrctl term
```

2. 컨테이너 내에서 다음 단계를 실행하십시오.

   a. **--force** 옵션으로 사용 가능한 미지원 부트스트랩 리포지토리를 나열합니다. 예제는 다음과 같습니다.

```
mgr-create-bootstrap-repo --list --force
1. SLE-12-SP2-x86_64
2. SLE-12-SP3-x86_64
```

   b. 다음과 같은 적절한 리포지토리 이름을 제품 레이블로 사용해 부트스트랩 리포지토리를 생성합니다.

```
mgr-create-bootstrap-repo --create SLE-12-SP2-x86_64 --force
```

부트스트랩 리포지토리를 수동으로 생성하고 싶지 않다면 필요한 제품 및 부트스트랩 리포지토리에 LTSS를 사용할 수 있는지 여부를 확인하시기 바랍니다.

29.3. 클라이언트 복제 Salt 클라이언트 문제 해결

하이퍼바이저 복제 유틸리티를 사용한 후 복제된 Salt 클라이언트를 등록하려 했다면 다음과 같은 오류가 표시될 수 있습니다.

```
죄송하지만 시스템을 찾을 수 없습니다.
```

이러한 오류의 원인은 복제된 새 시스템에 등록된 기존 시스템과 동일한 시스템 ID가 있기 때문입니다. 이 ID를 수동으로 변경하여 오류를 바로잡고 복제된 시스템을 등록할 수 있습니다.

추가 정보 및 지침은 **Administration › Troubleshooting**에서 확인할 수 있습니다.

29.4. 전체 디스크 이벤트 포함 컨테이너 문제 해결

컨테이너의 영구 스토리지 매체로 마운트된 전용 디스크의 스토리지 공간이 부족할 경우 긴급 조치가 필요합니다.

스토리지 미디어 크기 조정 문제를 해결하려면 다음을 수행합니다. 컨테이너 호스트에서 나열된 모든 명령을 **root** 권한으로 실행합니다.

절차: 스토리지 매체 크기 조정

1. 디스크 크기를 늘리십시오. 수행해야 할 작업은 설치 환경에 따라 다릅니다.
2. 디스크를 분할한 경우(예: 디스크 **/dev/vdb1**에 **/dev/vdb**가 있는 경우) 다음 명령을 실행합니다.

   다음 명령을 실행합니다.

   a. **parted /dev/vdb**

   b. **(parted) print**

   c. **(parted) resizepart NUMBER 100%**에서 **NUMBER**는 **print** 명령으로 표시되는 파티션 번호(예: **/dev/vdb1**인 경우 **1**)입니다.

   d. **(parted) quit**

3. 파일 시스템 크기를 조정합니다. 예를 들어, XFS 파일 시스템의 경우 다음 명령을 실행합니다.

```
xfs_growfs /dev/vdb1
```

절차를 완료하면 XFS 파일 시스템이 디스크에서 사용 가능한 모든 공간을 사용하고 있어야 합니다.

29.5. 손상된 리포지토리 문제 해결

리포지토리 메타데이터 파일의 정보가 손상되거나 만료될 수 있습니다. 이에 따라 클라이언트 업데이트에 문제가 발생할 수 있습니다. 파일을 제거하고 재생성하여 이 문제를 해결할 수 있습니다. 새 리포지토리 데이터 파일을 사용하면 업데이트가 예상대로 작동해야 합니다.

절차: 손상된 리포지토리 데이터 문제 해결하기

1. **/var/cache/rhn/repodata/<channel-label>**에서 모든 파일을 제거합니다. 채널 레이블을 모르는 경우 **소프트웨어 › 채널 › 채널 레이블**로 이동하여 SUSE Multi-Linux Manager Web UI에서 찾을 수 있습니다.
2. 컨테이너 호스트의 명령줄에서 다음 명령을 실행하여 컨테이너의 파일을 다시 생성합니다.

```
mgrctl exec -ti -- spacecmd softwarechannel_regenerateyumcache <channel-label>
```

29.6. 충돌하는 패키지가 포함된 사용자 정의 채널 문제 해결

부트스트랩 리포지토리 생성과 같은 충돌하는 패키지 기능으로 사용자 정의 채널을 설정할 때 정의되지 않은 동작이 발생하고 클라이언트 등록이 실패할 수 있습니다.

예를 들어, 버전 번호가 더 높은 충돌하는 패키지가 부트스트랩 리포지토리에 포함될 수 있습니다. 이러한 패키지(예: **python3-zmq** 또는 **zeromq**)는 부트스트랩 리포지토리의 생성을 손상시키거나 클라이언트 부트스트랩 중에 문제를 일으킬 수 있습니다.

When the custom channel (for example, an EPEL channel) is added below the parent vendor channel, issues with conflicting packages cannot be solved directly. The way how to solve this is to separate the custom channel from the vendor channel. The custom channel needs to be created in a separate tree. In case that the custom channel needs to be delivered as a child, such environment can be created using Content Lifecycle Management (CLM). Sources in a CLM project can be added there from different trees. Using such an approach, the custom channel stays below the parent within the built environment. However, the vendor channel tree stays without the custom channel and the bootstrap repository. Then registering clients works correctly.

충돌하는 패키지(salt, zeromq 등)가 포함된 사용자 정의 채널이 하위 채널로 생성되는 경우 다음 단계를 통해 문제를 방지할 수 있습니다.

절차: 사용자 정의 채널에서 패키지 충돌 방지

1. 상위 채널에서 하위 채널로 사용자 정의 채널을 제거합니다. 자세한 내용은 administration:custom-channels.pdf에서 확인할 수 있습니다.
2. 별도의 트리에 사용자 정의 채널을 생성합니다. 자세한 내용은 administration:custom-channels.pdf에서 확인할 수 있습니다.
3. CLM(컨텐트 라이프싸이클 관리)에서 사용자 정의 채널을 하위 채널로 임포트하려면 다음을 실행합니다.
   - SUSE Multi-Linux Manager Web UI에서 **컨텐트 라이프싸이클**로 이동하고 **[프로젝트 생성]**을 클릭합니다. **이름** 및 **레이블**을 입력합니다.
   - 프로젝트에 소스를 연결합니다. 필요한 벤더 채널과 사용자 정의 채널을 사용합니다. (CentOS8을 사용한 공유 예제)

- 프로젝트에 환경을 추가합니다. 예를 들어, CentOS8을 사용합니다.
- 환경을 빌드하려면 **[빌드]** 버튼을 클릭합니다. 이 작업을 수행하면 활성화 키와 연결되고 클라이언트를 부트스트랩하기 위해 사용할 수 있는 벤더 및 사용자 정의 채널이 포함된 환경이 생성됩니다.

4. 중요 참고: CLM 프로젝트에서, 문제가 있거나 충돌하는 패키지를 제외하는 필터를 추가하는 것이 좋습니다. 그렇지 않으면 클라이언트 업데이트 중에 버전 번호가 더 높은 충돌하는 패키지가 설치됩니다. 필터링에 대한 자세한 내용은 administration:content-lifecycle-examples.pdf에서 확인할 수 있습니다.
5. CLM 환경(벤더 및 사용자 정의 채널 포함)에 최신 패치를 패치하려면 프로젝트에서 **[빌드]** 버튼을 클릭합니다. 이는 환경을 다시 빌드하기 위해 필요합니다.
   - CLM에 대한 자세한 내용은 **Administration › Content-lifecycle**에서 확인할 수 있습니다.

Extra Packages for Enterprise Linux(EPEL)를 Red Hat Enterprise Linux 클라이언트(또는 호환 가능: SUSE Liberty Linux, CentOS, Oracle Linux 등)에서 직접 사용하면 EPEL의 Salt 패키지가 설치되며, 여기에는 SUSE Multi-Linux Manager 제공 Salt 패키지에서 제공되는 일부 기능이 누락됩니다. 이는 SUSE가 아닌 Salt 패키지가 포함된 부트스트랩 리포지토리를 생성하므로 중요한 사항입니다. 따라서 이 시나리오는 지원되지 않습니다.

EPEL 리포지토리를 활성화해야 하는 경우 미리 EPEL에서 Salt 패키지를 필터링해야 합니다(예: **소프트웨어 › 관리 › 채널 › EPEL › 패키지**에서 Salt 패키지 제거).

29.7. FQDNS 입자 비활성화 문제 해결

FQDNS grain은 시스템에 있는 모든 전체 DNS 서비스의 목록을 반환합니다. 이 정보를 수집하는 작업은 대개 빠르게 완료되지만 DNS 설정이 잘못 구성된 경우 더 많은 시간이 소요될 수 있습니다. 어떤 경우 클라이언트는 무응답 상태가 되거나 작동을 중지할 수 있습니다.

이러한 문제를 방지하려면 Salt 플래그로 FQDNS grain을 비활성화하면 됩니다. grain을 비활성화하면 네트워크 모듈을 사용해 클라이언트가 무응답 상태가 될 위험 없이 FQDNS 서비스를 제공할 수 있습니다.

이전 Salt 클라이언트에만 적용됩니다. 최근에 Salt 클라이언트를 등록했다면 FQDNS grain이 기본적으로 비활성화되어 있습니다.

SUSE Multi-Linux Manager 서버의 명령 프롬프트에서 다음 명령을 사용해 FQDNS grain을 비활성화합니다.

```
salt '*' state.sls util.mgr_disable_fqdns_grain
```

이 명령을 실행하면 각 클라이언트가 재시작되고 서버가 처리해야 하는 Salt 이벤트를 생성합니다. 클라이언트의 수가 매우 많은 경우 대신에 다음과 같이 배치 모드로 명령을 실행할 수 있습니다.

```
salt --batch-size 50 '*' state.sls util.mgr_disable_fqdns_grain
```

배치 명령이 실행을 완료할 때까지 기다립니다. `Ctrl` + `C`로 프로세스를 인터럽트하지 마십시오.

29.8. 디스크 공간 문제 해결

디스크 공간이 부족하면 SUSE Multi-Linux Manager 데이터베이스 및 파일 구조에 심각한 영향을 미칠 수 있으며,

대부분의 경우 이를 복구할 수 없습니다. SUSE Multi-Linux Manager는 특정 디렉토리의 여유 공간을 모니터링하고 구성 가능한 경고가 있습니다. 공간 관리에 대한 자세한 내용은 **Administration › Space-management**에서 확인할 수 있습니다.

일반적으로 컨테이너 볼륨은 호스트 파일 시스템과 공간을 공유합니다. **btrfs** 파일 시스템이 가득 차면 파일을 삭제하지 못할 수 있습니다. 이 문제를 해결하기 위해 **btrfs device add** 명령을 사용하여 파일 시스템에 소규모의 스토리지를 임시로 추가할 수 있습니다. 이러한 방식으로 파일을 삭제하여 추가 파일 시스템의 유지보수를 위한 공간을 확보할 수 있습니다. 유지보수가 완료되면 **btrfs device delete**로 임시 스토리지를 제거할 수 있습니다. 이 주제에 대한 자세한 내용은 https://www.suse.com/support/kb/doc/?id=000018779을 참조하십시오.

사용하지 않는 소프트웨어 채널을 제거하여 디스크 공간을 복구할 수 있습니다.

- 벤더 채널 삭제에 대한 내용은 **Administration › Custom-channels**에서 확인할 수 있습니다.
- 사용자 정의 채널을 삭제하는 방법에 대한 자세한 내용은 **Administration › Custom-channels**에서 확인할 수 있습니다.

사용자 정의 채널이 동기화되는 빈도도 확인할 수 있습니다. 사용자 정의 채널 동기화를 처리하는 방법에 대한 지침은 administration:custom-channels.pdf에서 확인할 수 있습니다.

사용하지 않는 활성화 키, 컨텐트 라이프싸이클 프로젝트 및 클라이언트 등록을 정리하여 디스크 공간을 복구할 수도 있습니다. 중복 데이터베이스 항목을 제거할 수도 있습니다.

절차: 중복 데이터베이스 항목 문제 해결하기

1. **spacewalk-data-fsck** 명령을 사용하여 중복 데이터베이스 항목을 나열합니다.
2. **spacewalk-data-fsck --remove** 명령을 사용하여 삭제합니다.

29.9. 방화벽 문제 해결

발신 트래픽을 차단하는 방화벽을 사용하는 경우 네트워크 요청이 **거부** 또는 **삭제**일 수 있습니다. **삭제**로 설정하면 SUSE Customer Center과(와) 동기화하는 시간이 초과될 수 있습니다.

이는 동기화 프로세스가 SUSE Customer Center뿐만 아니라 SUSE가 아닌 클라이언트를 위한 패키지를 제공하는 타사 리포지토리에 액세스해야 하기 때문에 발생합니다. SUSE Multi-Linux Manager 서버가 이러한 리포지토리에 도달하여 유효성 확인을 시도하면 방화벽이 요청을 삭제하고 동기화는 시간이 초과될 때까지 응답을 계속 기다립니다.

이 경우 동기화가 실패하기까지 오랜 시간이 걸리며 SUSE가 아닌 제품은 제품 목록에 표시되지 않습니다.

이 문제는 여러 방법으로 해결할 수 있습니다.

가장 간단한 방법은 SUSE가 아닌 리포지토리에 필요한 URL에 대한 액세스를 허용하도록 방화벽을 구성하는 것입니다. 이렇게 하면 동기화 프로세스가 URL에 도달하고 성공적으로 완료할 수 있습니다.

외부 트래픽을 허용할 수 없는 경우 **삭제** 대신 SUSE Multi-Linux Manager의 **거부** 요청으로 방화벽을 구성합니다. 이렇게 하면 타사 URL에 대한 요청이 거부되므로 동기화 시간이 초과되지 않고 조기에 실패하며 제품이 목록에 표시되지 않습니다.

방화벽에 구성 액세스 권한이 없는 경우 SUSE Multi-Linux Manager 서버에 별도의 방화벽을 대신 설정할 수 있습니다.

29.10. WAN 연결을 통한 SUSE Multi-Linux Manager 서버와 프록시 간의 긴 동기화 시간 문제 해결

WebUI에서 또는 배포 또는 시스템 설정에 대한 API 호출을 통해 실행되는 변경 사항에 따라, SUSE Multi-Linux Manager 서버에서 SUSE Multi-Linux Manager 프록시 시스템으로 파일을 전송하기 위해서는 **cobbler sync** 명령이 필요할 수 있습니다. 이를 위해 Cobbler는 **/etc/cobbler/settings**에 지정된 프록시 목록을 사용합니다.

설계상 **cobbler sync**는 변경되거나 최근에 추가된 파일만 동기화하는 것이 불가능합니다.

대신, **cobbler sync**를 실행하면 **/srv/tftpboot** 디렉토리를 **/etc/cobbler/settings**에 구성된 모든 지정된 프록시로 전체 동기화합니다. 또한 관련 시스템 간의 WAN 연결 대기 시간에 의해서도 영향을 받습니다.

/var/log/cobbler/의 로그에 따라, 동기화 프로세스를 완료하는 데 상당한 시간이 걸릴 수 있습니다.

예를 들어, 다음에 시작됨:

```
Thu Jun  3 14:47:35 2021 - DEBUG | running python triggers from
/var/lib/cobbler/triggers/task/sync/pre/*
Thu Jun  3 14:47:35 2021 - DEBUG | running shell triggers from
/var/lib/cobbler/triggers/task/sync/pre/*
```

및 다음에 종료됨:

```
Thu Jun  3 15:18:49 2021 - DEBUG | running shell triggers from
/var/lib/cobbler/triggers/task/sync/post/*
Thu Jun  3 15:18:49 2021 - DEBUG | shell triggers finished successfully
```

전송량은 약 1.8GB였습니다. 전송에 약 30분이 걸렸습니다.

이에 비해 **/srv/tftboot**와 같은 크기의 대용량 단일 파일 복사는 몇 분 내에 완료됩니다.

SUSE Multi-Linux Manager 서버와 프록시 간에 파일을 복사하는 **rsync** 기반 접근 방식으로 전환하면 전송 및 대기 시간을 줄이는 데 도움이 될 수 있습니다.

이 작업을 수행하기 위한 스크립트는 https://suse.my.salesforce.com/sfc/p/1i000000gLOd/a/1i000000ll5B/B2AmvIJN2_JsAyjTQzCVP_x5ioVgd0bYN9X9NpMugS8에서 다운로드할 수 있습니다.

스크립트는 명령줄 옵션을 승인하지 않습니다. 스크립트를 실행하기 전, 스크립트를 수동으로 편집하고 올바르게 작동하려면 **MLMHOSTNAME**, **MLMIP** 및 **MLMPROXY1** 변수를 올바르게 설정해야 합니다.

스크립트의 개별 조정에 사용할 수 있는 지원은 제공되지 않습니다. 스크립트와 내부 주석은 고려해야 할 프로세스 및 단계에 대한 개요를 제공하는 것을 목표로 합니다. 추가적인 도움이 필요한 경우 SUSE Consulting에 문의하십시오.

스크립트를 사용하는 제안된 접근법은 다음 환경에서 유용합니다.

- SUSE Multi-Linux Manager 프록시 시스템은 WAN 연결을 통해 연결됩니다.
- **/srv/tftboot**에 배포용 파일 및 클라이언트 PXE 부팅 파일이 많이 포함되어 있으며, 전체 파일이 수천 개 있습니다.

- **/etc/cobbler/settings**에 있는 프록시가 비활성화되었습니다. 그렇지 않으면, SUSE Multi-Linux Manager은(는) 프록시에 컨텐트를 계속 동기화합니다.

```
#프록시:
# - "MLMproxy.MLMproxy.test"
# - "MLMproxy2.MLMproxy.test"
```

절차: 새로운 동기화 속도 분석

1. SUSE Multi-Linux Manager 및 관련 시스템 간의 TCP 트래픽 덤프를 가져옵니다.
   - SUSE Multi-Linux Manager 서버에서:

     ```
     tcpdump -i ethX -s 200 host <ip-address-of-susemanagerproxy> and not ssh
     ```

   - SUSE Multi-Linux Manager 프록시에서:

     ```
     tcpdump -i ethX -s 200 host <ip-address-of-susemanager> and not ssh
     ```

   - 그러면 분석을 실행하기에 충분한 200개의 패키지 크기만 캡처됩니다.
   - SUSE Multi-Linux Manager가 프록시와의 통신에 사용하는 각 네트워크 인터페이스에 대한 ethX를 조정합니다.
   - 마침내, 더 이상 패키지 수를 줄이기 위해 ssh 통신이 캡처되지 않습니다.
2. **cobbler sync**를 시작합니다.
   - 동기화를 강제 실행하려면, Cobbler json 캐시 파일을 먼저 삭제한 후 **cobbler sync**를 실행합니다.

     ```
     rm /var/lib/cobbler/pxe_cache.json
     cobbler sync
     ```

3. **cobbler sync**가 완료되면 TCPdump를 중지합니다.
4. Wireshark를 사용하여 TCPdump를 열고 **통계 > 대화**로 이동하여 덤프가 분석될 때까지 기다립니다.
5. TCP 탭으로 전환합니다. 이 탭에 표시되는 숫자는 SUSE Multi-Linux Manager 서버와 SUSE Multi-Linux Manager 프록시 사이에서 수집된 총 대화 수를 나타냅니다.
6. **기간** 열 찾기:
   - 파일 전송에 소요된 최소 시간을 확인하려면 우선 오름차순으로 정렬합니다.
   - 내림차순으로 계속 정렬하여, 예를 들어 커널 및 initrd 전송과 같은 대용량 파일의 최대값을 찾습니다.

     포트 4505 및 4506은 Salt 통신에 사용되므로 무시합니다.

TCP 덤프를 분석한 결과 SUSE Multi-Linux Manager 서버에서 프록시로 약 1,800바이트 크기의 작은 파일을 전송하는 데 약 0.3초가 걸렸습니다.

대용량 파일이 많지는 않았지만, 전송된 모든 단일 파일에 대해 새로운 TCP 연결이 생성되기 때문에 작은 파일의 수가 많으면 많은 수의 연결이 설정됩니다.

그러므로 최소 전송 시간과 필요한 연결 수(예시에서는 약 5,000개)를 알면 전체 전송 시간에 대한 대략적인 예상 시간이 5000 * 0.3 / 60 = 25분이라는 것을 알 수 있습니다.

29.11. 비활성 클라이언트 문제 해결

Taskomatic 작업은 주기적으로 클라이언트를 ping하여 연결 여부를 확인합니다. 24시간 이상 Taskomatic 체크인에 응답하지 않는 클라이언트는 비활성 상태로 간주됩니다. Web UI에서 비활성 클라이언트 목록을 보려면 **시스템 › 시스템 목록 › 비활성**으로 이동합니다.

클라이언트는 여러 이유로 비활성화될 수 있습니다.

- 클라이언트가 SUSE Multi-Linux Manager 서비스에 대한 자격이 부여되지 않았습니다.
- HTTPS 연결을 허용하지 않는 방화벽의 후방에 클라이언트가 위치합니다.
- 클라이언트가 잘못 구성된 프록시의 후방에 위치합니다.
- 클라이언트가 다른 SUSE Multi-Linux Manager 서버와 통신 중이거나 연결이 잘못 구성되었습니다.
- 클라이언트가 SUSE Multi-Linux Manager 서버와 통신할 수 있는 네트워크에 없습니다.
- 방화벽이 클라이언트와 SUSE Multi-Linux Manager 서버 간의 트래픽을 차단 중입니다.
- Taskomatic이 잘못 구성되었습니다.

서버로의 클라이언트 연결에 대한 자세한 내용은 **Client-configuration › Contact-methods-intro**에서 확인할 수 있습니다.

포트 구성에 대한 자세한 내용은 installation-and-upgrade:network-requirements.pdf에서 확인할 수 있습니다.

방화벽 문제 해결에 대한 자세한 내용은 **Administration › Troubleshooting**에서 확인할 수 있습니다.

29.12. 서버 간 동기화 문제 해결

서버 간 동기화는 캐시를 사용하여 ISS 마스터 및 슬레이브를 관리합니다. 이러한 캐시에서는 잘못된 항목을 생성하는 버그가 발생하기 쉽습니다. 이 경우 캐시가 여전히 잘못된 항목을 사용하고 있기 때문에 버그를 해결하는 버전으로 업데이트한 후에도 버그가 발생할 수 있습니다. ISS의 새 버전으로 업그레이드했지만, 여전히 문제가 발생하면 모든 캐시를 제거하여 문제가 발생하는 이전 항목이 모두 제거되어야 합니다.

캐시 오류는 다양한 오류와 함께 동기화 실패로 이어질 수 있지만, 오류 메시지는 일반적으로 다음과 같은 항목을 보고합니다.

```
/var/cache/rhn/satsync/*에서 satellite-sync 캐시를 제거하고 동일한 옵션으로 satellite-sync를 다시 실행합니다.
```

ISS 마스터 및 ISS 슬레이브에서 캐시를 삭제하여 이 문제를 해결할 수 있으므로 동기화가 성공적으로 완료됩니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

절차: ISS 캐싱 오류 해결

1. ISS 마스터의 명령 프롬프트에서 루트 권한으로 마스터에 대한 캐시 파일을 삭제합니다.

```
rm -rf /var/cache/rhn/xml-*
```

2. 서비스를 다시 시작합니다.

```
rcapache2 restart
```

3. ISS 마스터의 명령 프롬프트에서 루트 권한으로 슬레이브에 대한 캐시 파일을 삭제합니다.

```
rm -rf /var/cache/rhn/satsync/*
```

4. 서비스를 다시 시작합니다.

```
rcapache2 restart
```

29.13. 로컬 발급자 인증서 문제 해결

일부 이전 부트스트랩 스크립트는 잘못된 위치에 로컬 인증서에 대한 링크를 생성합니다. 그 결과, zypper가 로컬 발급자 인증서에 대해 **인식할 수 없는 오류**를 반환합니다. **/etc/ssl/certs/** 디렉토리를 확인하여 로컬 발급자 인증서에 대한 링크가 올바르게 생성되었는지 확인할 수 있습니다. 이러한 문제가 발생하면 zypper가 올바르게 작동하도록 부트스트랩 스크립트를 업데이트하는 것을 고려해야 합니다.

29.14. 로그인 시간 제한 문제 해결

기본적으로 SUSE Multi-Linux Manager Web UI에서는 사용자가 30 분 후에 다시 로그인해야 합니다. 환경에 따라 로그인 시간 제한 값을 조정할 수 있습니다.

값을 조정하려면 **rhn.conf** 및 **web.xml**을 모두 변경해야 합니다. **/etc/rhn/rhn.conf**에서는 값을 초 단위로 설정하고 **web.xml**에서는 분 단위로 값을 설정해야 합니다. 두 값은 시간의 양이 동일해야 합니다.

예를 들어, 제한 시간 값을 한 시간으로 변경하려면 **rhn.conf**의 값을 3600초로, **web.xml**의 값을 60분으로 설정합니다.

절차: Web UI 로그인 시간 제한 값 조정

1. 컨테이너 호스트에서, 서버 컨테이너 내의 명령줄을 엽니다.

```
mgrctl term
```

   a. **/etc/rhn/rhn.conf**를 열고 새로운 시간 제한 값(초)을 포함하도록 다음 라인을 추가하거나 편집합니다.

```
web.session_database_lifetime = <Timeout_Value_in_Seconds>
```

   b. 파일을 저장한 후 닫습니다.

   c. **/etc/tomcat/web.xml**를 열고 새로운 시간 제한 값(분)을 포함하도록 다음 라인을 추가하거나 편집합니다.

```
<session-timeout>Timeout_Value_in_Minutes</session-timeout>
```

d. 파일을 저장한 후 닫습니다.

2. On the container host, restart the server to enforce the new configuration:

```
systemctl restart uyuni-server.service
```

29.15. 메일 구성 문제 해결

보안 메일 통신을 위해 인증을 활성화하고, 사용자 이름과 비밀번호를 정의한 후, **/etc/rhn/rhn.conf**에서 **SSL** 또는 **STARTLS**를 활성화합니다.

```
java.smtp_server = string (default: localhost)
java.smtp_port = integer (default: 25)
java.smtp_auth = true/false (default: false)
java.smtp_ssl = true/false (default: false)
java.smtp_starttls = true/false (default: false)
java.smtp_user = string (default: null)
java.smtp_pass = string (default: null)
```

SMTP 서버 통신의 연결 시간 제한을 늘리려면 **/etc/rhn/rhn.conf**에서 다음 파라미터를 설정할 수 있습니다.

```
java.smtp_timeout = integer (default: 5000)
java.smtp_connection_timeout = integer (default: 5000)
java.smtp_write_timeout = integer (default: 5000)
```

29.16. 대량 Machine_id 중복

SUSE Multi-Linux Manager를 사용하여 가상 머신을 관리하는 경우 VM의 클론을 생성하는 것이 유용할 수 있습니다. 클론은 기존 디스크의 복사본인 기본 디스크를 사용하는 VM입니다.

VM을 복제하면 시간을 크게 절약할 수 있지만, 디스크에 중복된 식별 정보가 있으면 문제가 발생할 수 있습니다.

두 대의 컴퓨터를 등록하려고 할 때 한 컴퓨터가 다른 컴퓨터의 복제본인 경우, SUSE Multi-Linux Manager은(는) 이를 별개의 두 클라이언트로 등록하고자 할 수 있습니다. 그러나 원본 클라이언트와 복제본 모두의 컴퓨터 ID가 동일한 경우, SUSE Multi-Linux Manager은(는) 두 클라이언트를 하나의 시스템으로 등록하며, 두 번째 컴퓨터가 등록되면 데이터가 덮어써집니다.

이 문제는 복제본의 머신 ID를 변경하여 SUSE Multi-Linux Manager이(가) 두 개의 서로 다른 클라이언트로 인식하도록 하면 해결할 수 있습니다. 머신 ID 변경 방법에 대한 자세한 정보와 지침은 **Administration › Troubleshooting**에서 확인할 수 있습니다.

하지만 일부 경우에는 기존 머신 ID에 의존하는 다른 애플리케이션에 장애를 유발할 수 있으므로, 모든 시스템에서 머신 ID를 변경할 수 없으며 이것이 실용적이지도 않습니다. 이러한 경우 SUSE Multi-Linux Manager 환경에서 사용하기 위해 가짜 머신 ID를 생성할 수 있는 옵션이 제공됩니다.

절차: SUSE Multi-Linux Manager의 등록 시점에 머신 ID 생성

1. 부트스트랩 스크립트를 생성합니다(이미 있는 경우 이 단계를 건너뜀). 자세한 내용은 **Client-configuration › Registration-bootstrap**에서 확인할 수 있습니다.
2. 파일을 열고 파라미터를 변경합니다.

```
GENERATE_OWN_MACHINEID=1
```

3. 부트스트랩을 엄격하게 사용하면 머신 ID가 더 이상 충돌하지 않습니다.

29.17. noexec를 사용한 /tmp 마운트 문제 해결

Salt는 클라이언트 파일 시스템의 **/tmp**에서 원격 명령을 실행합니다. 그러므로 **noexec** 옵션으로 **/tmp**를 마운트하지 않아야 합니다. 이 문제를 해결하기 위한 다른 방법은 **noexec** 옵션이 설정되지 않은 디렉토리를 가리키도록 Salt 서비스 지정된 **TMPDIR** 환경 변수로 임시 디렉터리 경로를 재정의하는 것입니다. 클라이언트에서 Salt 번들을 사용하는 경우 systemd 드롭인 구성 파일 **/etc/systemd/system/venv-salt-minion.service.d/10-TMPDIR.conf**를 사용하거나 **salt-minion**을 사용하는 경우에는 **/etc/systemd/system/salt-minion.service.d/10-TMPDIR.conf**를 사용하는 것이 좋습니다. 드롭인 구성 파일 내용의 예:

```
[Service]
Environment=TMPDIR=/var/tmp
```

29.18. noexec를 사용한 /var/tmp 마운트 문제 해결

Salt SSH는 **/var/tmp**를 사용하여 Salt Bundle을 배포하고 번들된 Python으로 클라이언트에서 Salt 명령을 실행합니다. 그러므로 **noexec** 옵션으로 **/var/tmp**를 마운트하지 않아야 합니다. 부트스트랩 프로세스는 Salt SSH를 사용하여 클라이언트에 도달하므로 **/var/tmp**가 **noexec** 옵션으로 마운트된 클라이언트를 Web UI로 부트스트랩할 수 없습니다.

29.19. 디스크 공간 부족 문제 해결

사용할 수 있는 디스크 공간을 확인한 후 마이그레이션을 시작하십시오. 별도의 XFS 파일 시스템에서 **/var/spacewalk** 및 **/var/lib/pgsql**을 찾는 것이 좋습니다.

별도의 파일 시스템을 설정하는 경우에는 **/etc/fstab**을 편집하고 **/var/lib/pqsql** 하위 볼륨을 제거하십시오. 서버를 재부팅하여 변경 사항을 적용하십시오.

업그레이드 문제에 대한 자세한 정보는 마이그레이션 로그 파일을 확인하십시오. 로그 파일은 업그레이드 중인 시스템의 **/var/log/rhn/migration.log**에 있습니다.

29.20. 알림 문제 해결

알림 메시지의 기본 수명은 30일이며, 그 이후에는 읽기 상태와 관계없이 메시지가 데이터베이스에서 삭제됩니다. 이 값을 변경하려면 **/etc/rhn/rhn.conf**에서 다음 라인을 추가하거나 편집합니다.

```
java.notifications_lifetime = 30
```

알림 유형을 활성화 또는 비활성화하려면 **/etc/rhn/rhn.conf**에서 다음 라인을 추가하거나 편집합니다.

```
java.notifications_type_disabled = OnboardingFailed,ChannelSyncFailed,\
  ChannelSyncFinished,CreateBootstrapRepoFailed,StateApplyFailed,\
  PaygAuthenticationUpdateFailed,EndOfLifePeriod,SubscriptionWarning
```

기본 설정 및 구성 옵션은 **usr/share/rhn/config-defaults/rhn_java.conf** 템플릿 파일을 참조하십시오.

29.21. OES 리포지토리 활성화 문제 해결

SUSE Multi-Linux Manager 서버에서 OES(Open Enterprise Server)를 활성화하려면 설명되는 절차를 따릅니다.

절차: OES 리포지토리 활성화

1. OES에 액세스할 수 있는 Microfocus의 미러 자격 증명이 있는지 확인합니다.
2. SUSE Multi-Linux Manager 서버에 로그인합니다.
3. **관리자 > 설정 마법사 > 조직 자격 증명**으로 이동합니다.
4. SUSE Multi-Linux Manager에 대한 SUSE 자격 증명이 이미 있는지 확인합니다.
5. 새로 추가하기 위한 옵션을 선택하고 Microfocus 자격 증명을 입력합니다.
6. **관리자 > 설정 마법사 > 조직 자격 증명**으로 이동하여 새로 고침 작업이 완료될 때까지 기다립니다.
7. 새로 고친 제품 목록에 OES가 표시되어야 합니다. 이제 다른 제품처럼 활성화할 수 있습니다.

OES에 대한 자세한 내용은 https://www.microfocus.com/documentation/open-enterprise-server/에서 확인할 수 있습니다.

29.22. 패키지 불일치 문제 해결

클라이언트의 패키지가 잠긴 경우 SUSE Multi-Linux Manager 서버가 적용 가능한 패치 집합을 올바르게 결정하지 못할 수 있습니다. 이 경우 Web UI에서 패키지 업데이트를 사용할 수 있지만, 클라이언트에는 표시되지 않으며 클라이언트 업데이트 시도가 실패합니다. 패키지 잠금 및 제외 목록을 확인하여 클라이언트에서 패키지가 잠겨 있는지 또는 제외되는지 확인합니다.

클라이언트에서 패키지 잠금 및 제외 목록을 확인하여 패키지가 잠긴 상태인지 또는 제외되었는지 확인합니다.

- 확장 지원 플랫폼에서 **/etc/yum.conf**를 확인하고 **exclude=**를 검색합니다.
- SUSE Linux Enterprise 및 openSUSE에서 **zypper locks** 명령을 사용합니다.

29.23. 시작 이벤트로 입자 전달 문제 해결

Salt 클라이언트는 시작할 때마다 **machine_id** grain을 SUSE Multi-Linux Manager로 전달합니다. SUSE Multi-Linux Manager는 이 grain을 사용해 클라이언트가 등록되었는지 여부를 판별합니다. 이 프로세스를 진행하려면 동기 Salt 호출이 필요합니다. 동기 Salt 호출은 다른 프로세스를 차단하므로 다수의 클라이언트가 동시에 시작하는 경우 프로세스가 상당히 지연될 수 있습니다.

이 문제를 해결하기 위해 별도의 동기 Salt 호출을 방지할 수 있는 새로운 기능이 Salt에 도입되었습니다.

이 기능을 사용하려면 이 기능을 지원하는 클라이언트에서 클라이언트 구성에 구성 파라미터를 추가하면 됩니다.

이 프로세스가 더 원활히 진행되게 하려면 **mgr_start_event_grains.sls** 도우미 Salt 상태를 사용하면 됩니다.

이전에 등록한 클라이언트에만 적용됩니다. 최근에 Salt 클라이언트를 등록했다면 이 구성 파라미터가 기본적으로 추가되어 있습니다.

SUSE Multi-Linux Manager 서버의 명령 프롬프트에서 다음 명령을 사용해 **start_event_grains** 구성 도우미를 활성화합니다.

```
salt '*' state.sls util.mgr_start_event_grains
```

이 명령은 필요한 구성을 클라이언트의 구성에 추가하고, 클라이언트를 재시작할 때 이를 적용합니다. 클라이언트가 수가 매우 많은 경우 대신에 다음과 같이 배치 모드로 명령을 실행할 수 있습니다.

```
salt --batch-size 50 '*' state.sls mgr_start_event_grains
```

29.24. 프록시 연결 및 FQDN 문제 해결

프록시를 통해 연결된 클라이언트가 Web UI에 표시되지만 프록시를 통해 연결되었음을 표시하지 않는 경우가 있습니다. 연결에 전체 도메인 이름(FQDN)을 사용하지 않고 프록시가 SUSE Multi-Linux Manager에 알려지지 않은 경우 이러한 문제가 발생할 수 있습니다.

이러한 동작을 교정하려면 다음과 같이 프록시의 클라이언트 구성 파일에서 추가 FQDN을 grain으로 지정하십시오.

```
grains:
  susemanager:
     custom_fqdns:
       - name.one
       - name.two
```

29.25. 문제 해결 복제된 클라이언트 등록

SUSE Multi-Linux Manager를 사용하여 가상 머신을 관리하는 경우 VM의 클론을 생성하는 것이 유용할 수 있습니다. 클론은 기존 디스크의 복사본인 기본 디스크를 사용하는 VM입니다.

VM을 복제하면 시간을 크게 절약할 수 있지만, 디스크에 중복된 식별 정보가 있으면 문제가 발생할 수 있습니다.

이미 등록된 클라이언트가 있는 경우 해당 클라이언트의 클론을 생성한 후 클론을 등록하고, SUSE Multi-Linux Manager를 두 개의 개별 클라이언트로 등록하고자 할 수 있습니다. 그러나 원본 클라이언트와 클론의 머신 ID가 동일한 경우 SUSE Multi-Linux Manager는 두 클라이언트를 하나의 시스템으로 등록하고 기존 클라이언트 데이터를 클론의 데이터로 덮어씁니다.

이는 SUSE Multi-Linux Manager가 클론을 서로 다른 클라이언트 두 개로 인식하도록 클론의 머신 ID를 변경하면 해결할 수 있습니다.

이 절차의 각 단계는 복제된 클라이언트에서 수행됩니다. 이 절차는 SUSE Multi-Linux Manager에 등록된 상태로 남아 있는 원래 클라이언트를 조작하지 않습니다.

절차: 복제된 Salt 클라이언트에서 중복 머신 ID 해결

1. **Initial System Configuration**

   a. 복제된 머신에서 호스트 이름과 IP 주소를 변경합니다. **/etc/hosts**에 변경사항과 올바른 호스트 항목이 포함되어 있는지 확인합니다.

2. **Resolving Duplicate Machine IDs**

   a. systemd를 지원하는 배포판의 경우:

      i. 머신의 머신 ID가 동일한 경우, 복제된 각 클라이언트에서 루트 권한으로 파일을 삭제하고 다시 생성합니다.

```
rm /etc/machine-id
rm /var/lib/dbus/machine-id
rm /var/lib/zypp/AnonymousUniqueId
dbus-uuidgen --ensure
systemd-machine-id-setup
```

      ii. If the cloned machine also has a folder in **/var/log/journal/** it needs to be renamed accordingly to the new machine ID. If names do not match, **journalctl** could not retrieve any log and **podman logs** would not show anything.

```
mv /var/log/journal/* /var/log/journal/$(cat /etc/machine-id)
```

   b. systemd를 지원하지 않는 배포판의 경우:

      i. 루트 권한으로 dbus에서 머신 ID를 생성합니다.

```
rm /var/lib/dbus/machine-id
rm /var/lib/zypp/AnonymousUniqueId
dbus-uuidgen --ensure
```

- 나중에 SUSE Liberty Linux로 전환될 Red Hat Enterprise Linux 8.10 서버를 복제하는 경우, 커널 구성 파일을 수정하기 위한 추가 단계를 수행해야 합니다.
- Red Hat Enterprise Linux는 머신 ID를 사용하여 **/boot/loader/entries**에 커널 항목을 생성합니다. 이 단계를 수행하지 않으면 전환 후 SUSE Liberty Linux 커널이 기존 항목을 대체하지 않고 새 항목을 생성하므로, 기존 및 새 커널

항목이 혼재된 상태가 됩니다.

- 머신 ID 변경 후 전환하기 전에 다음 명령을 실행하십시오.

```
sudo rm -rf /boot/loader/entries/
sudo for ver in $(rpm -q kernel --qf '%{VERSION}-
%{RELEASE}.%{ARCH}\n'); do echo "Reinstalling kernel
$ver..."; sudo kernel-install add $ver /lib/modules/$ver;
done
sudo grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

- Red Hat Enterprise Linux 8.10 서버의 보안 부팅 해제에 대한 자세한 정보와 예시는 **Common-workflows › Workflow-liberate-rhel-with-secureboot**에서 확인할 수 있습니다.

3. **Reconfiguring Salt Clients**

   a. If your clients still have the same Salt client ID, delete the **minion_id** file on each client (FQDN is used when it is regenerated on client restart).

      i. For Salt Minion clients:

```
rm /etc/salt/minion_id
rm -rf /etc/salt/pki
```

      ii. Salt 번들 클라이언트의 경우 다음과 같습니다.

```
rm /etc/venv-salt-minion/minion_id
rm -rf /etc/venv-salt-minion/pki
```

   b. Delete accepted keys from the onboarding page and the system profile from SUSE Multi-Linux Manager, and restart the client with.

      i. For Salt Minion clients:

```
service salt-minion restart
```

      ii. Salt 번들 클라이언트의 경우 다음과 같습니다.

```
service venv-salt-minion restart
```

   c. 클라이언트를 다시 등록합니다. 이제 각 클라이언트는 서로 다른 **/etc/machine-id**를 가지며, **시스템 개요** 페이지에 올바르게 표시되어야 합니다.

29.26. SL Micro의 원격 루트 로그인

보안 강화를 위해 SL Micro 6.1 이상을 새로 설치할 때 비밀번호 기반 원격 루트 로그인은 더 이상 허용되지 않으며, 이는 SL Micro에서 실행되는 서버 및 프록시 컨테이너 호스트와 관리형 SL Micro 클라이언트에도 영향을 미칩니다. 또한, 비밀번호 기반 원격 루트 로그인이 설정된 SLE Micro 5.5 클라이언트는 6.1/6.2로 마이그레이션될 때 이 접근 권한이 갑자기 제거되며, 새로 구성해야 합니다. 자세한 내용은 SL Micro 릴리스 노트 6.1(https://www.suse.com/releasenotes/x86_64/SL-Micro/6.1/index.html#jsc-SMO-405)을 참조하십시오.

SUSE Multi-Linux Manager 프록시와 같은 SUSE Multi-Linux Manager 구성 요소를 배포하는 경우에는 기본적으로 비밀번호 기반 원격 루트 로그인이 필요합니다. 다음 단계를 통해 비밀번호 기반 원격 루트 로그인을 활성화할 수 있습니다.

절차: SL Micro에서 비밀번호를 사용한 SSH 루트 로그인 활성화

두 가지 방법으로 SSH 루트 로그인을 활성화할 수 있습니다. 두 가지 방법 중 가장 적합한 방법을 선택하면 됩니다.

옵션 A: 사전 구성 패키지 사용

1. 클라이언트의 UI/API에서 **openssh-server-config-rootlogin** 패키지를 설치합니다.
2. 컨테이너 호스트를 재부팅해 UI 또는 터미널에서 새 구성을 활성화합니다.

옵션 B: SSH 구성을 수작업으로 편집

1. 내용이 다음과 같은 드롭인 구성 파일 **/etc/ssh/sshd_config.d/permit_root.conf**를 추가합니다.

```
PermitRootLogin yes
```

2. SSH 서버 구성 재로드

```
systemctl reload sshd
```

3. SSH로 연결된 경우, 서버에서 연결을 끊기 전에 새 SSH 연결을 열어 SSH 서버가 정상적으로 작동하는지 확인합니다.

transactional-update에 대한 자세한 내용은 https://documentation.suse.com/sle-micro/6.1/html/Micro-transactional-updates/에서 확인할 수 있습니다.

29.27. 삭제된 클라이언트 등록 문제 해결

서버 컨테이너 내에서 단계를 실행하기 전에 **mgrctl term**을 사용합니다.

삭제된(등록되지 않은) 클라이언트를 새로 등록하지 못할 수 있습니다. 이 문제를 해결하려면 SUSE Multi-Linux Manager 서버(Salt 마스터)에서 일부 Salt 캐시 파일을 삭제한 후 다시 등록해야 합니다.

```
rm /var/cache/salt/master/thin/version
rm /var/cache/salt/master/thin/thin.tgz
```

29.28. 문제 해결 Web UI에서 등록이 실패하고 오류가 표시되지 않음

Web UI에서의 초기 등록을 위해 모든 Salt 클라이언트는 Salt SSH를 사용합니다.

기본적으로 Salt SSH 클라이언트는 서버에 오류를 다시 보고하지 않습니다.

그러나 Salt SSH 클라이언트는 오류를 검사할 수 있는 **/var/log/salt-ssh.log**에 로그를 로컬로 저장합니다.

29.29. 문제 해결 Red Hat CDN 채널 및 다중 인증서

Red Hat 컨텐트 제공 네트워크(CDN) 채널은 경우에 따라 여러 인증서를 제공하지만 SUSE Multi-Linux Manager Web UI는 한 개의 인증서만 임포트할 수 있습니다. CDN이 SUSE Multi-Linux Manager Web UI에서 알고 있는 인증서와 다른 인증서를 제공하는 경우, 인증서가 올바르더라도 유효성 검사에 실패하고 리포지토리에 대한 액세스가 거부됩니다. 수신되는 오류 메시지는 다음과 같습니다.

```
[error]
Repository '<repo_name>' is invalid.
<repo.pem> Valid metadata not found at specified URL
History:
 - [|] Error trying to read from '<repo.pem>'
 - Permission to access '<repo.pem>' denied.
이 리포지토리에 대해 정의된 URI가 유효한 리포지토리를 가리키고 있는지 확인하십시오.
위의 오류로 인해 '<repo_name>' 리포지토리를 건너뛰었습니다.
오류로 인해 리포지토리를 새로 고칠 수 없습니다.
HH:MM:SS RepoMDError: 리포지토리에 액세스할 수 없습니다. 리포지토리 GPG 키를 가져오지 못했을
수 있습니다.
```

이 문제를 해결하려면 유효한 모든 인증서를 단일 **.pem** 파일로 병합한 후 SUSE Multi-Linux Manager에서 사용하도록 인증서를 다시 빌드하십시오.

절차: 여러 Red Hat CDN 인증서 해결

1. Red Hat 클라이언트의 명령 프롬프트에서, 루트 권한으로 단일 **rh-cert.pem** 파일의 **/etc/pki/entitlement/**에서 모든 현재 인증서를 수집합니다.

   ```
   cat 866705146090697087.pem 3539668047766796506.pem redhat-entitlement- authority.pem
   > rh-cert.pem
   ```

2. 단일 **rh-key. pem** 파일의 **/etc/pki/entitlement/**에서 모든 현재 키를 수집합니다.

   ```
   cat 866705146090697087-key.pem 3539668047766796506-key.pem > rh-key.pem
   ```

Client-configuration › Clients-rh-cdn의 지침에 따라 이제 새 인증서를 SUSE Multi-Linux Manager 서버로 임포트할 수 있습니다.

29.30. SUSE Multi-Linux Manager 서버 이름 변경 문제 해결

SUSE Multi-Linux Manager 서버의 호스트 이름을 로컬로 변경하면 SUSE Multi-Linux Manager 설치가 제대로 작동하지 않습니다. 왜냐하면 데이터베이스에서 변경사항이 적용되지 않아 변경 사항이 클라이언트와 프록시로 전파되지 않기 때문입니다.

29.30.1. Rename server

SUSE Multi-Linux Manager 서버의 호스트 이름을 변경해야 하는 경우 **mgradm server rename** 명령을 사용하여 변경할 수 있습니다. 이 명령은 PostgreSQL 데이터베이스의 설정 및 SUSE Multi-Linux Manager의 내부 구조를 업데이트합니다.

29.30.1.1. Server configuration

이 명령에는 필수 파라미터가 없지만, 컨테이너 호스트의 호스트 이름이 아닌 경우 새 호스트 이름을 지정할 수 있습니다.

새 호스트 이름과 일치하도록 SSL 인증서를 생성해야 하는 경우 SSL CA 비밀번호를 제공해야 합니다. 이 작업은 구성 파일을 사용하여 안전하게 수행할 수 있습니다.

절차: SSL CA 비밀번호의 구성 파일 준비

1. 다음과 같이 **config.yaml** 파일을 작성합니다.

```
ssl:
  password: "<CA PASSWORD>"
```

This file will be used in the next procedure.

절차: SUSE Multi-Linux Manager 서버 이름 변경

1. DNS 서버에서 로컬 및 원격으로 시스템 수준에서 서버의 네트워크 설정을 변경합니다. 역방향 이름 확인을 위한 구성 설정도 제공해야 합니다. 네트워크 설정 변경은 다른 시스템의 이름을 변경할 때와 동일한 방식으로 수행됩니다.
2. 새 네트워크 구성을 사용하고 호스트 이름이 변경되었는지 확인하려면 SUSE Multi-Linux Manager 서버를 재부팅합니다.
3. 컨테이너 호스트에서 명령줄을 통해 다음 명령을 실행하십시오. SSL CA 비밀번호를 저장하기 위해 파일을 생성한 경우 **-c config.yaml**을 추가합니다.

```
mgradm server rename
```

   새 호스트 이름이 확인되지 않으면 명령이 실패합니다.

서버 컨테이너의 재시작 중에도 이름 바꾸기 절차가 수행됩니다. 다음 명령을 실행하면 로그를 확인할 수 있습니다.

```
mgrctl exec -ti -- journalctl -u uyuni-update-config
```

서버 컨테이너를 재시작할 때 이 명령은 모든 Salt 클라이언트의 열 데이터의 새로 고침을 트리거합니다: 실행 시간은 등록된 클라이언트 수에 따라 달라집니다.

29.30.1.2. Directly managed clients reconfiguration

Skip this procedure if clients are managed via a SUSE Multi-Linux Manager proxy.

다음 절차에 따라 직접 관리 클라이언트가 새 호스트 이름과 IP 주소를 인식하도록 재구성합니다.

Procedure: Reconfiguring directly managed clients

1. 모든 클라이언트의 Salt 클라이언트 구성 파일에서 새 Salt 마스터(SUSE Multi-Linux Manager 서버)의 이름을 지정합니다. 파일 이름은 **/etc/venv-salt-minion/minion.d/susemanager.conf** 또는 Salt 번들을 사용하지 않는 경우 **/etc/salt-minion/minion.d/susemanager.conf**입니다.

   ```
   master: <new_hostname>
   ```

2. 모든 클라이언트에서 Salt 서비스를 다시 시작합니다. 다음 중 하나를 실행합니다.

   ```
   systemctl restart venv-salt-minion
   ```

   또는 Salt 번들을 사용하지 않는 경우에는 다음을 실행합니다.

   ```
   systemctl restart salt-minion
   ```

29.30.1.3. Client connection with applying high state

마지막으로 호스트 이름을 Salt 클라이언트 구성에 완전히 전파하려면 highstate를 적용합니다. highstate를 적용하면 리포지토리 URL의 호스트 이름이 업데이트됩니다.

29.30.2. Reconfigure proxy

모든 프록시를 다시 구성해야 합니다. 새 서버 인증서와 키를 프록시에 복사해야 합니다. 자세한 내용은 **Installation-and-upgrade › Install-proxy**에서 확인할 수 있습니다.

프록시를 통해 PXE 부팅을 사용하는 경우 프록시의 구성 설정을 확인해야 합니다. 컨테이너화되지 않은 SUSE Multi-Linux Manager 프록시 4.3을 통해 PXE 부팅을 사용하는 경우, **tftpsync**를 재구성해야 합니다.

컨테이너 호스트에서 다음을 실행합니다.

```
mgrctl exec -ti -- configure-tftpsync.sh
```

29.31. 문제 해결 RPC 연결 시간 제한

네트워크 속도가 느리거나 네트워크 링크가 다운되어 RPC 연결이 시간 제한에 도달할 수 있습니다. 이에 따라 패키지 다운로드 또는 배치 작업이 중단되거나 예상보다 오래 걸립니다. 구성 파일을 편집하여 RPC 연결에 소요될 수 있는 최대 시간을 조정할 수 있습니다. 이 작업을 수행해도 네트워킹 문제가 해결되지는 않지만, 프로세스는 중단되지 않고 실패하게 됩니다.

절차: RPC 연결 시간 제한 확인

1. SUSE Multi-Linux Manager 서버에서 **/etc/rhn/rhn.conf** 파일을 열고 최대 시간 제한 값(초)을 설정합니다.

```
server.timeout =`number`
```

2. SUSE Multi-Linux Manager 프록시에서 **/etc/uyuni/proxy/config.yaml** 파일을 열고 최대 시간 초과 값(초 단위)을 설정하십시오. 변경 사항이 적용되려면 프록시 컨테이너를 다시 시작해야 합니다.

```
timeout =`number`
```

3. zypper를 사용하는 SUSE Linux Enterprise Server 클라이언트에서 **/etc/zypp/zypp.conf** 파일을 열고 최대 시간 제한 값(초)을 설정합니다.

```
## Valid values:  [0,3600]
## Default value: 180
download.transfer_timeout = 180
```

4. yum을 사용하는 Red Hat Enterprise Linux 클라이언트에서 **/etc/yum.conf** 파일을 열고 최대 시간 제한 값(초)을 설정합니다.

```
timeout =`number`
```

RPC 시간 제한을 **180** 초 미만으로 제한하면, 완전히 정상적인 작업이 중단될 위험이 있습니다.

29.32. 문제 해결 다운 및 DNS 설정으로 표시된 Salt 클라이언트

Salt 클라이언트가 실행 중인 경우에도 패키지 새로 고침 또는 적용 상태 등의 작업이 실패로 표시되며 다음 메시지가 표시됩니다.

```
Minion이 작동 중지되거나 연결할 수 없습니다.
```

이 경우 작업 일정을 변경해 보십시오. 일정을 변경할 수 있는 경우 잘못된 DNS 구성이 문제의 원인일 수 있습니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

Salt 클라이언트가 다시 시작되거나 그레인이 새로 고쳐지는 경우, 클라이언트는 FQDN 그레인을 계산하고 그레인이 진행될 때까지 응답하지 않습니다. SUSE Multi-Linux Manager 서버에서 예약된 작업이 실행될 때 SUSE Multi-Linux Manager 서버는 클라이언트를 대상으로 **test.ping**을 수행한 후 실제 작업을 수행하여 클라이언트가 실제로 실행되고 있고 작업이 트리거될 수 있는지 확인합니다.

기본적으로 SUSE Multi-Linux Manager 서버는 **test.ping** 명령의 응답을 수신하기 위해 5초 동안 대기합니다. 5초 이내에 응답이 수신되지 않으면 클라이언트가 다운되었거나 연결할 수 없다는 메시지를 표시하고 작업이 실패하도록 설정됩니다.

이 문제를 해결하려면 클라이언트에서 DNS 확인을 수정하여 FQDN을 확인하는 동안 클라이언트가 5초 동안 멈추지 않도록 하십시오.

가능하지 않은 경우 SUSE Multi-Linux Manager 서버의 **/etc/rhn/rhn.conf** 파일에 있는 **java.salt_presence_ping_timeout** 값을 4보다 큰 값으로 증가하십시오.

예:

```
mgrctl term
vim /etc/rhn/rhn.conf
java.salt_presence_ping_timeout = 6
```

이후에 다음을 실행합니다.

```
mgradm restart
```

이 값을 증가시키면 SUSE Multi-Linux Manager 서버가 미니언에 연결할 수 없거나 응답하지 않는지 여부를 확인하는 데 시간이 더 오래 걸릴 수 있으므로, SUSE Multi-Linux Manager 서버가 전반적으로 느려지거나 응답하지 않습니다.

29.33. 스키마 업그레이드 실패 문제 해결

스키마 업그레이드에 실패하면 데이터베이스 버전 확인 및 기타 모든 Spacewalk 서비스가 시작되지 않습니다. 자세한 정보와 진행 방법에 대한 힌트는 컨테이너 호스트에서 **mgradm start**를 실행하여 확인할 수 있습니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

컨테이너에서 직접 버전 확인을 실행할 수도 있습니다.

```
systemctl status uyuni-check-database.service
```

또는

```
journalctl -u uyuni-check-database.service
```

이러한 명령은 더 일반적인 **mgradm** 명령을 실행하지 않으려는 경우 디버그 정보를 출력합니다.

29.34. 문제 해결 동기화

동기화는 여러 이유로 실패할 수 있습니다. 연결 문제에 대한 자세한 내용을 확인하려면 다음 명령을 실행합니다.

```
export URLGRABBER_DEBUG=DEBUG
spacewalk-repo-sync -c <channelname> <options> > /var/log/spacewalk-repo-sync-$(date +%F-
%R).log 2>&1
```

Zypper가 생성한 로그는 **/var/log/zypper.log**에서도 확인할 수 있습니다.

GPG 키 불일치

SUSE Multi-Linux Manager는 타사 GPG 키를 자동으로 신뢰하지 않습니다. 패키지 동기화가 실패하면 신뢰할 수 없는 GPG 키 때문일 수 있습니다. **/var/log/rhn/reposync**를 열고 다음과 같은 오류를 찾으면 이러한 경우에 해당하는지 확인 가능:

```
['/usr/bin/spacewalk-repo-sync', '--channel', 'sle-12-sp1-ga-desktop-
nvidia-driver-x86_64', '--type', 'yum', '--non-interactive']
RepoMDError: Cannot access repository. Maybe repository GPG keys are not imported
```

문제를 해결하려면 GPG 키를 SUSE Multi-Linux Manager로 임포트해야 합니다. GPG 키 임포트에 대한 자세한 내용은 **Administration › Repo-metadata**에서 확인할 수 있습니다.

spacewalk-repo-sync에서 GPG 키 제거

When a GPG key for repository has been manually imported using **spacewalk-repo-sync**, and this key is no longer needed (for example if the key was compromised, or was used for testing purposes only), it can be removed from the zypper RPM database used by **spacewalk-repo-sync** with the following command:

```
rpm --dbpath=/var/lib/spacewalk/reposync/root/var/lib/rpm/ -e gpg-pubkey-*
```

여기서 **gpg-pubkey-***는 제거할 GPG 키의 이름입니다.

GPG 키 갱신

GPG 키를 갱신하려면 먼저 이전 키를 제거한 후 새 키를 생성하여 임포트합니다.

체크섬 불일치

체크섬이 실패한 경우 **/var/log/rhn/reposync/*.log** 로그 파일에 다음과 같은 오류가 표시될 수 있습니다.

```
Repo Sync Errors: (50, u'checksums did not match
326a904c2fbd7a0e20033c87fc84ebba6b24d937 vs
afd8c60d7908b2b0e2d95ad0b333920aea9892eb', 'Invalid information uploaded
to the server')
The package microcode_ctl-1.17-102.57.62.1.x86_64 which is referenced by
patch microcode_ctl-8413 was not found in the database. This patch has
been skipped.
```

이 오류는 **-Y** 옵션을 사용하여 명령 프롬프트에서 동기화를 실행하여 해결 가능합니다.

```
spacewalk-repo-sync --channel <channelname> -Y
```

이 옵션은 로컬로 캐시된 체크섬이 아닌 동기화 전에 리포지토리 데이터를 확인합니다.

연결 시간 제한

다운로드 시간이 다음 오류와 함께 초과하는 경우에 해당합니다.

```
28, '작업이 너무 느립니다. 최근 300초 내에 전송된 초당 1000바이트 미만
```

이 오류는 **/etc/rhn/rhn.conf**에서 **reposync_timeout** 및 **reposync_minrate** 구성 값을 지정하여 해결할 수 있습니다. 기본적으로 300초 동안 초당 1000바이트 미만이 전송되면 다운로드가 중단됩니다. **reposync_minrate**로 초당 바이트 수를 조정하고 **reposync_timeout**으로 대기 시간(초)을 조정할 수 있습니다.

재동기화 중 수동으로 키 신뢰하기

경우에 따라 **reposync**를 실행할 때 GPG 키를 수동으로 수락해야 할 수 있습니다. 예:

```
# spacewalk-repo-sync -c nvidia-compute-sle-15-x86_64-we-sp3
17:07:40 ======================================
17:07:40 | Channel: nvidia-compute-sle-15-x86_64-we-sp3
17:07:40 ======================================
17:07:40 Sync of channel started.
수신된 새 리포지토리 또는 패키지 서명 키:
  리포지토리:       nvidia-compute-sle-15-x86_64-we-sp3
  키 지문:  610C 7B14 E068 A878 070D A4E9 9CD0 A493 D42D 0685
  키 이름:         cudatools <cudatools@nvidia.com>
  키 알고리즘:    RSA 4096
  키 생성 시간:      Thu Apr 14 16:04:01 2022
  키 만료:      (만료 없음)
  Rpm 이름:         gpg-pubkey-d42d0685-62589a51
    참고: 데이터 서명을 통해 수신자는 데이터 서명 후 변경이 발생하지 않았음을 확인할 수
있습니다.
    서명이 없거나 잘못되었거나 알 수 없는 서명이 있는 데이터를 수락하면 시스템 손상이 발생할
수 있으며,
    극단적인 경우 시스템 침해로 이어질 수도 있습니다.
    참고: GPG 공개 키는 지문으로 명확하게 식별됩니다. 키 이름에 의존하지 마십시오.
    제공된 키의 진위 여부가 확실하지 않다면 리포지토리 제공자에게 문의하거나 해당 웹사이트를
확인하십시오.
    많은 제공자는 사용 중인 GPG 키의 지문을 보여주는 웹 페이지를 제공합니다.
키를 거부하시겠습니까, 일시적으로 신뢰하시겠습니까, 아니면 항상 신뢰하시겠습니까? [r/t/a/?]
(r):
```

29.35. 문제 해결 Taskomatic

리포지토리 메타데이터 재생성은 비교적 집약적인 프로세스이므로 Taskomatic을 완료하려면 몇 분이 걸릴 수 있습니다. 또한, Taskomatic이 충돌하면 리포지토리 메타데이터 재생성이 중단될 수 있습니다.

서버 컨테이너 내부의 셸에 액세스하려면 컨테이너 호스트에서 **mgrctl term**을 실행합니다.

Taskomatic이 여전히 실행 중이거나 프로세스가 충돌한 경우 패키지 업데이트가 Web UI에서 사용할 수 있는 것처럼 보일 수 있지만, 클라이언트에는 표시되지 않으며 클라이언트 업데이트 시도가 실패합니다. 이 경우 **zypper ref** 명령에서 표시되는 오류는 다음과 같습니다.

```
지정된 URL에서 유효한 메타데이터를 찾을 수 없음
```

이를 수정하려면 Taskomatic이 여전히 리포지토리 메타데이터를 생성하는 중인지 또는 충돌이 발생할 수 있는지 확인합니다. 클라이언트 업데이트가 올바르게 수행되도록 메타데이터 재생성이 완료될 때까지 기다리거나 충돌 후 Taskomatic을 재시작합니다.

절차: Taskomatic 문제 확인

1. SUSE Multi-Linux Manager 서버에서 **/var/log/rhn/rhn_taskomatic_daemon.log** 파일을 확인하여 메타데이터 재생성 프로세스가 아직 실행 중인지 또는 충돌이 발생했는지 확인합니다.
2. taskomatic을 재시작합니다.

```
service taskomatic restart
```

3. Taskomatic 로그 파일에서 다음과 같은 여는 라인 및 닫는 라인을 찾아 메타데이터 재생성과 관련된 섹션을 식별할 수 있습니다.

```
<YYYY-DD-MM> <HH:MM:SS>,174 [Thread-584] INFO
com.redhat.rhn.taskomatic.task.repomd.RepositoryWriter - Generating new repository
metadata for channel 'cloned-2018-q1-sles12-sp3-updates-x86_64'(sha256) 550 packages,
140 errata

...

<YYYY-DD-MM> <HH:MM:SS>,704 [Thread-584] INFO
com.redhat.rhn.taskomatic.task.repomd.RepositoryWriter - Repository metadata
generation for 'cloned-2018-q1-sles12-sp3-updates-x86_64' finished in 4 seconds
```

29.36. Web UI 로드 실패 문제 해결

가끔 마이그레이션 후에 Web UI가 로드되지 않을 수 있습니다. 이 문제는 새 시스템의 호스트 이름 및 IP 주소가 기존 시스템과 동일한 경우 주로 브라우저 캐싱에 의해 발생합니다. 이러한 중복으로 일부 브라우저에서 혼동이 발생할 수 있습니다.

캐시를 지우고 페이지를 다시 로드하면 이 문제가 해결됩니다. 대부분의 브라우저에서는 `Ctrl` + `F5`를 눌러 이 문제를 빠르게 해결할 수 있습니다.

Chapter 30. GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections

then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum

below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this

License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
   Permission is granted to copy, distribute and/or modify this document
   under the terms of the GNU Free Documentation License, Version 1.2
   or any later version published by the Free Software Foundation;
   with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
   A copy of the license is included in the section entitled "GNU
   Free Documentation License".
```