

Uyuni 2026.01

설치 및 업그레이드 가이드



U Y U N I

장 1. 서문

Installation, Deployment and Upgrade + Uyuni 2026.01

This guide provides comprehensive, step-by-step instructions for deploying, upgrading, and managing Uyuni Server and Proxy.

It is organized into the following sections:

- **Requirements:** Outlines the essential hardware, software, and networking prerequisites to ensure a smooth setup.
 - **Deployment and Installation:** Guides you through deploying Uyuni as a container and completing the initial configuration.
 - **Upgrade and Migration:** Details the process for upgrading and migrating Uyuni while minimizing downtime.
 - **Basic Server Management:** Covers fundamental server operations, helping you get started with Uyuni efficiently.

게시 날짜: 2026-01-28

† † † † † † † † † † † † † † † † †

차례

| | |
|--|-----------|
| 1. 서문 | 1 |
| 2. 요구사항 | 3 |
| 2.1. 일반 요구사항 | 3 |
| 2.1.1. 서버 요구사항 | 3 |
| 2.1.2. 프록시 요구사항 | 3 |
| 2.2. 네트워크 요구사항 | 4 |
| 2.2.1. FQDN(정규화된 도메인 이름) | 4 |
| 2.2.2. 호스트 이름 및 IP 주소 | 5 |
| 2.2.3. Reenable router advertisements | 5 |
| 2.2.4. Deployment behind HTTP or HTTPS OSI level 7 proxy | 6 |
| 2.2.5. air-gapped 배포 | 7 |
| 2.2.6. 필수 네트워크 포트 | 7 |
| 2.3. 공용 클라우드 요구사항 | 12 |
| 2.3.1. 네트워크 요구사항 | 13 |
| 2.3.2. 스토리지 볼륨 준비 | 13 |
| 3. 배포 및 설치 | 15 |
| 3.1. Uyuni 서버 설치 | 15 |
| 3.1.1. Uyuni Server Deployment on openSUSE Tumbleweed | 15 |
| 3.1.2. Uyuni 서버 air-gapped 배포 | 18 |
| 3.2. Install Uyuni Proxy | 19 |
| 3.2.1. 컨테이너화된 Uyuni 프록시 설정 | 19 |
| 3.2.2. Uyuni Proxy Deployment on openSUSE Tumbleweed | 23 |
| 3.2.3. 클라이언트에서의 프록시 변환 | 28 |
| 3.2.4. K3s에 Uyuni 프록시 배포 | 31 |
| 4. 업그레이드 및 마이그레이션 | 33 |
| 4.1. 서버 | 33 |
| 4.1.1. Migrating the Uyuni Server to openSUSE Tumbleweed | 33 |
| 4.1.2. 레거시 Uyuni 서버를 컨테이너로 마이그레이션 | 36 |
| 4.1.3. Uyuni 서버 업그레이드 | 40 |
| 4.2. 프록시 | 41 |
| 4.2.1. Migrating the Uyuni Proxy to openSUSE Tumbleweed | 41 |
| 4.2.2. 레거시 프록시를 컨테이너로 마이그레이션 | 45 |
| 4.2.3. Uyuni 프록시 업그레이드 | 48 |
| 4.3. 클라이언트 | 49 |
| 4.3.1. 클라이언트 업그레이드 | 49 |
| 5. Basic Server and Proxy Management | 50 |
| 5.1. Custom YAML Configuration and Deployment with mgradm | 50 |
| 5.2. 컨테이너 시작 및 중지 | 51 |
| 5.3. Containers used by Uyuni | 51 |
| 5.4. 영구 컨테이너 볼륨 | 52 |
| 5.4.1. 서버 | 52 |
| 5.4.2. 프록시 | 54 |
| 5.5. mgr-storage-server 및 mgr-storage-proxy 이해하기 | 54 |
| 5.5.1. 이러한 도구의 기능 | 54 |
| 5.5.2. 이러한 도구가 제공하지 않는 기능 | 55 |
| 5.5.3. 설치 후 저장소 관리 | 55 |
| 5.5.4. 사용해야 하는 경우 또는 사용하지 않아야 하는 경우 | 56 |
| 5.5.5. Summary | 56 |
| 6. GNU Free Documentation License | 57 |

장 2. 요구사항

2.1. 일반 요구사항

다음 테이블은 최소 서버 및 프록시 요구사항을 지정합니다.



- Do not use NFS for storage because it does not support SELinux file labeling.

2.1.1. 서버 요구사항

표 1. x86-64 아키텍처에 대한 서버 요구사항

| Software and Hardware | Details | Recommendation |
|-----------------------|--------------------------------|---|
| Tumbleweed | Clean installation, up-to-date | Tumbleweed |
| CPU | - | Minimum 4 dedicated 64-bit CPU cores (x86-64) |
| RAM | Test or Base Installation | Minimum 16 GB |
| | Production Server | Minimum 32 GB |
| Disk Space | / (root directory) | Minimum 40 GB |
| | /var/lib/pgsql | Minimum 50 GB |
| | /var/spacewalk | Minimum storage required: 100 GB (this will be verified by the implemented check) * 각 SUSE 제품 및 Package Hub당 50 GB 각 Red Hat 제품당 360GB |
| | /var/cache | 최소 10 GB. SUSE 제품당 100MB, Red Hat 또는 기타 제품당 1GB를 추가합니다. 서버가 ISS 마스터인 경우 공간을 두 배로 늘립니다. |
| | 공간 스왑 | 3GB |

2.1.2. 프록시 요구사항

표 2. 프록시 요구사항

| Software and Hardware | Details | Recommendation |
|-----------------------|--------------------------------|----------------|
| Tumbleweed | Clean installation, up-to-date | Tumbleweed |

| Software and Hardware | Details | Recommendation |
|-----------------------|--------------------|--------------------------------------|
| CPU | | Minimum 2 dedicated 64-bit CPU cores |
| RAM | Test Server | Minimum 2 GB |
| | Production Server | Minimum 8 GB |
| Disk Space | / (root directory) | Minimum 40 GB |
| | /srv | Minimum 100 GB |
| | /var/cache (Squid) | Minimum 100 GB |

Uyuni 프록시는 **/var/cache/** 디렉토리에 패키지를 캐시합니다. **/var/cache/**의 공간이 부족한 경우 프록시는 사용되지 않는 오래된 패키지를 제거한 후 새 패키지로 교체합니다.

이 동작의 결과:

- 프록시에서 **/var/cache/** 디렉토리에 더 많은 공간이 확보되고 프록시와 Uyuni 서버 간의 트래픽이 감소합니다.
- 프록시에서 **/var/cache/** 디렉토리의 크기와 Uyuni 서버에서 **/var/spacewalk/**의 크기를 동일하게 설정하면, 최초 동기화 후 대규모 트래픽이 발생하는 것을 방지할 수 있습니다.
- Uyuni 서버의 **/var/cache/** 디렉토리는 프록시에 비해 작을 수 있습니다. 크기 예상에 대한 설명은 [\[server-hardware-requirements\]](#) 섹션을 참조하십시오.

2.2. 네트워크 요구사항

이 섹션에서는 Uyuni의 네트워크 및 포트 요구사항에 대한 자세한 설명을 제공합니다.

IP forwarding will be enabled by containerized installation. This means Uyuni Server and Proxies will behave as a router. This behavior is done by podman directly. Podman containers do not run if IP forwarding is disabled.



Consider achieving network isolation of the Uyuni environment according to your policies.

For more information, see <https://www.suse.com/support/kb/doc/?id=000020166>.

2.2.1. FQDN(정규화된 도메인 이름)

Uyuni 서버는 FQDN이 올바르게 확인되어야 합니다. FQDN을 확인할 수 없는 경우 여러 다른 구성 요소에서 심각한 문제가 발생할 수 있습니다.

호스트 이름 및 DNS 구성에 대한 자세한 내용은 <https://documentation.suse.com/sles/15-SP6/html/SLES-all/cha-network.html#sec-network-yast-change-host>에서 확인할 수 있습니다.

2.2.2. 호스트 이름 및 IP 주소

클라이언트가 Uyuni 도메인 이름을 올바르게 확인하도록 하려면, 서버 및 클라이언트 머신 모두 작동하는 DNS 서버에 연결되어야 합니다. 또한, 역방향 조회도 올바르게 구성되었는지 확인해야 합니다.

DNS 서버 설정에 대한 자세한 내용은 <https://documentation.suse.com/sles/15-SP6/html/SLES-all/cha-dns.html>에서 확인할 수 있습니다.

2.2.3. Reenable router advertisements

When the Uyuni is installed using **mgradm install podman** or **mgrpky install podman**, it sets up Podman which enables IPv4 and IPv6 forwarding. This is needed for communication from the outside of the container.

However, if your system previously had **/proc/sys/net/ipv6/conf/eth0/accept_ra** set to **1**, it will stop using router advertisements. As a result, the routes are no longer obtained via router advertisements and the default IPv6 route is missing.

To recover correct functioning of the IPv6 routing, follow the procedure depending on whether:

- server and proxy are based on 15 SP7 (without Network manager)
- server and proxy are based on SL Micro 6.1} (with Network manager)

Procedure: Reenabling router advertisements without Network Manager

1. Create a file in **/etc/sysctl.d**, for example **99-ipv6-ras.conf**.
2. 다음 파라미터와 값을 파일에 추가:

```
net.ipv6.conf.eth0.accept_ra = 2
```

3. 재부팅합니다.

Procedure: Reenabling router advertisements with Network Manager

1. List your connections with **nmcli connection show**.
2. Create or modify the file **/etc/NetworkManager/system-connections/<name of connection>.nmconnection** to add this setting:

```
[ipv6]
addr-gen-mode=eui64
```

3. 재부팅합니다.
4. The file should look similar to this:

```
[connection]
```

```

id=Wired connection 1
type=ethernet
interface-name=eth0

[ethernet]

[ipv4]
dns-priority=20
method=auto

[ipv6]
addr-gen-mode=eui64
method=auto

```

2.2.4. Deployment behind HTTP or HTTPS OSI level 7 proxy

Some environments enforce internet access through a HTTP or HTTPS proxy. This could be a Squid server or similar. To allow the Uyuni Server internet access in such configuration, you need to configure the following.

Procedure: Configuring HTTP or HTTPS OSI level 7 proxy

- For operating system internet access, modify `/etc/sysconfig/proxy` according to your needs:

```

PROXY_ENABLED="no"
HTTP_PROXY=""
HTTPS_PROXY=""
NO_PROXY="localhost, 127.0.0.1"

```

- For `Podman` container internet access, modify `/etc/systemd/system/uyuni-server.service.d/custom.conf` according to your needs. For example, set:

```

[Service]
Environment=TZ=Europe/Berlin
Environment="PODMAN_EXTRA_ARGS="
Environment="https_proxy=user:password@http://192.168.10.1:3128"

```

- For Java application internet access, modify `/etc/rhn/rhn.conf` according to your needs. On the container host, execute `mgrctl term` to open a command line inside the server container:

- Modify `/etc/rhn/rhn.conf` according to your needs. For example, set:

```

# Use proxy FQDN, or FQDN:port
server.satellite.http_proxy =
server.satellite.http_proxy_username =
server.satellite.http_proxy_password =
# no_proxy is a comma seperated list
server.satellite.no_proxy =

```

4. On the container host, restart the server to enforce the new configuration:

```
systemctl restart uyuni-server.service
```

2.2.5. air-gapped 배포

내부 네트워크에 위치하고 SUSE Customer Center에 액세스할 수 없는 경우, **Installation-and-upgrade > Container-deployment**을 사용할 수 있습니다.

In a production environment, the Uyuni Server and clients should always use a firewall. For a comprehensive list of the required ports, see [installation-and-upgrade:network-requirements.pdf](#).

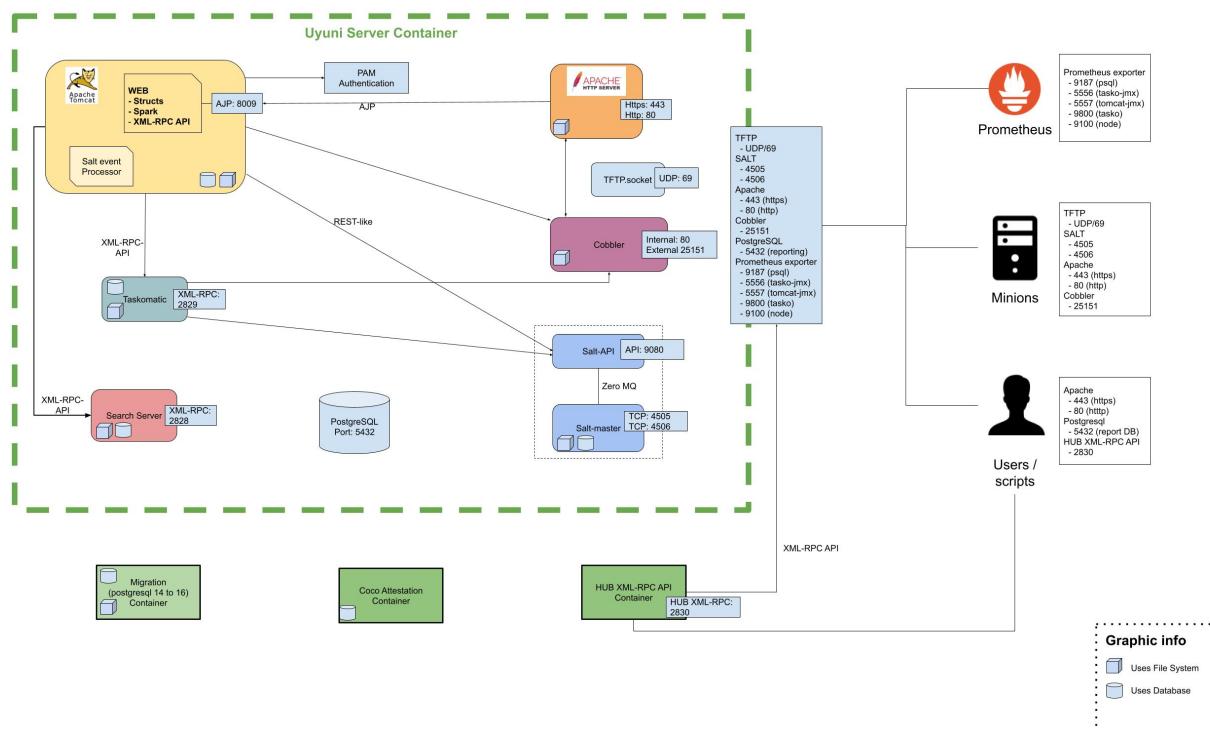
2.2.6. 필수 네트워크 포트

이 섹션에서는 Uyuni에서의 다양한 통신을 위해 사용되는 전체 포트 목록이 제공됩니다.

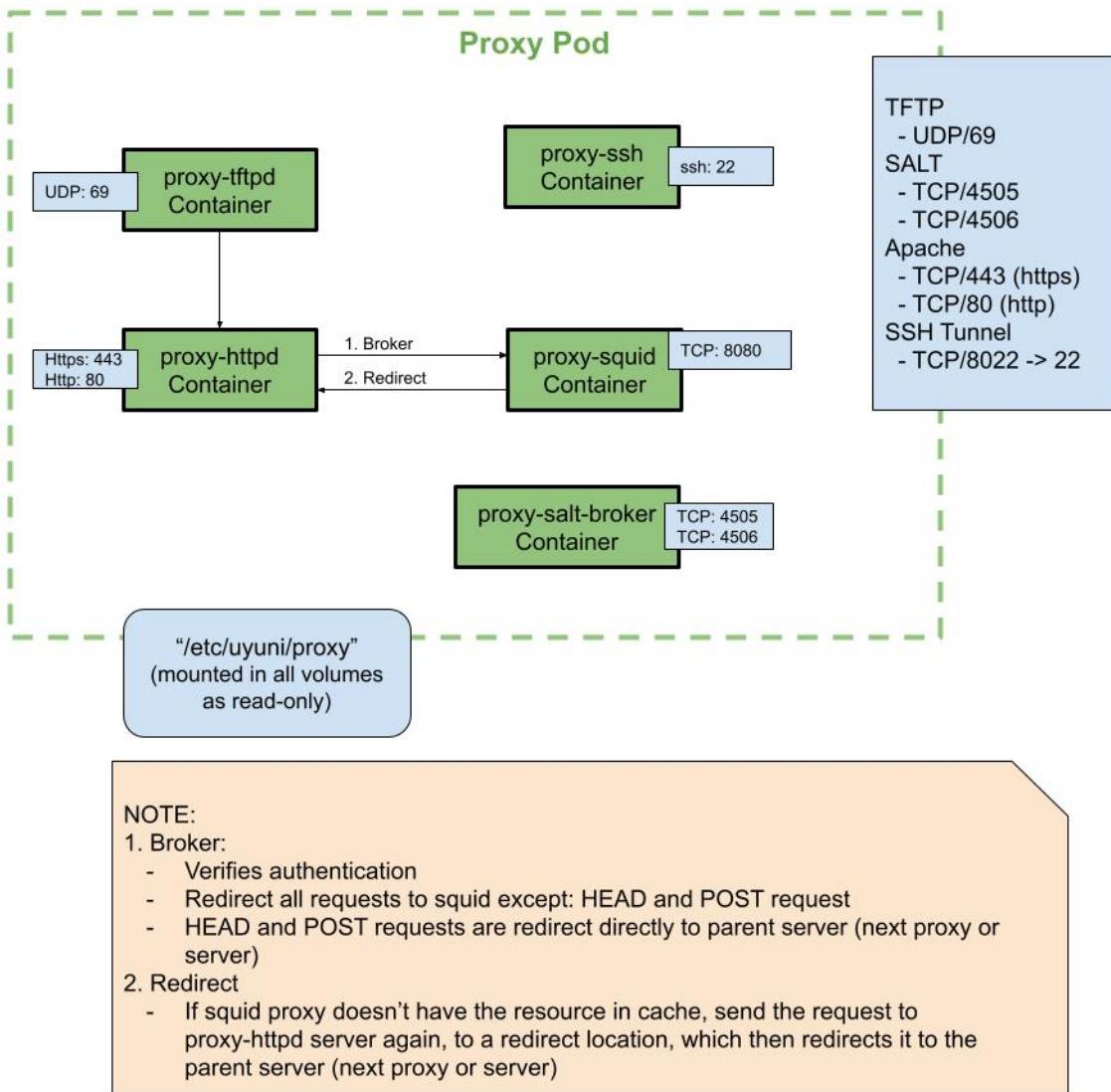
이러한 모든 포트를 열 필요는 없습니다. 사용 중인 서비스에 필요한 포트만 열면 됩니다.

2.2.6.1. Overview

2.2.6.1.1. 서버



2.2.6.1.2. 프록시



2.2.6.2. 외부 인바운드 서버 포트

무단 액세스로부터 서버를 보호하려면 Uyuni 서버에서 외부 인바운드 포트를 열어 방화벽을 구성해야 합니다.

이러한 포트를 열면 외부 네트워크 트래픽이 Uyuni 서버에 액세스할 수 있습니다.

표 3. Uyuni 서버의 외부 포트 요구사항

| 포트 번호 | 프로토콜 | 사용 대상 | 참고 |
|-------|---------|-------|---------------------------------------|
| 67 | TCP/UDP | DHCP | 클라이언트가 서버의 IP 주소를 요청할 때만 필요합니다. |
| 69 | TCP/UDP | TFTP | 자동 클라이언트 설치를 위해 서버가 PXE로 사용될 때 필요합니다. |
| 80 | TCP | HTTP | 일부 부트스트랩 리포지토리 및 자동 설치에서 일시적으로 필요합니다. |

| 포트 번호 | 프로토콜 | 사용 대상 | 참고 |
|-------|------|------------|--|
| 443 | TCP | HTTPS | Web UI, 클라이언트, 서버 및 프록시(tftpsync) 요청입니다. |
| 4505 | TCP | salt | 클라이언트로부터의 통신 요청을 수락하기 위해 필요합니다. 클라이언트가 연결을 시작하며, Salt 마스터로부터 명령을 수신하기 위해 열려 있는 상태를 유지합니다. |
| 4506 | TCP | salt | 클라이언트로부터의 통신 요청을 수락하기 위해 필요합니다. 클라이언트가 연결을 시작하며, Salt 마스터로 결과를 다시 보고하기 위해 열려 있는 상태를 유지합니다. |
| 5432 | TCP | PostgreSQL | 보고 데이터베이스에 액세스하는 데 필요합니다. |
| 5556 | TCP | Prometheus | Taskomatic JMX 메트릭 스크래핑에 필요합니다. |
| 5557 | TCP | Prometheus | Tomcat JMX 메트릭 스크래핑에 필요합니다. |
| 9100 | TCP | Prometheus | 노드 익스포터 메트릭 스크래핑에 필요합니다. |
| 9187 | TCP | Prometheus | PostgreSQL 메트릭 스크래핑에 필요합니다. |
| 9800 | TCP | Prometheus | Taskomatic 메트릭 스크래핑에 필요합니다. |
| 25151 | TCP | Cobbler | |

2.2.6.3. 외부 아웃바운드 서버 포트

액세스할 수 있는 서버를 제한하려면 Uyuni 서버에서 외부 아웃바운드 포트를 열어 방화벽을 구성해야 합니다.

이러한 포트를 열면 Uyuni 서버로부터의 네트워크 트래픽이 외부 서비스와 통신할 수 있습니다.

표 4. Uyuni 서버의 외부 포트 요구사항

| Port number | Protocol | Used By | Notes |
|-------------|----------|---------|---|
| 80 | TCP | HTTP | Required for SUSE Customer Center. Port 80 is not used to serve the Web UI. |
| 443 | TCP | HTTPS | Required for SUSE Customer Center. |
| 25151 | TCP | Cobbler | |

2.2.6.4. 내부 서버 포트

내부 포트는 Uyuni 서버에 의해 내부적으로 사용됩니다. 내부 포트는 **localhost**에서만 액세스할 수 있습니다.

대부분의 경우에는 이러한 포트를 조정할 필요가 없습니다.

표 5. Uyuni 서버의 내부 포트 요구사항

| 포트 번호 | 참고 |
|-------|---|
| 2828 | Satellite-search API, Tomcat 및 Taskomatic의 RHN 애플리케이션에서 사용됩니다. |
| 2829 | Taskomatic API, Tomcat의 RHN 애플리케이션에서 사용됩니다. |
| 8005 | Tomcat 종료 포트입니다. |
| 8009 | Tomcat-Apache HTTPD(AJP)입니다. |
| 8080 | Tomcat-Apache HTTPD(HTTP)입니다. |
| 9080 | Salt-API, Tomcat 및 Taskomatic의 RHN 애플리케이션에서 사용됩니다. |
| 25151 | Cobbler의 XMLRPC API |
| 32000 | Taskomatic 및 satellite-search에서 실행되는 Java 가상 머신(JVM)으로의 TCP 연결을 위한 포트입니다. |

32768 이상 포트는 사용 후 삭제 포트로 사용됩니다. 이러한 포트는 대부분 TCP 연결을 수신하기 위해 사용됩니다. TCP 연결 요청이 수신되면, 발신자가 이러한 사용 후 삭제 포트 번호 중 하나를 선택하여 대상 포트로 사용합니다.

다음 명령을 사용하여 임시 포트인 포트를 찾을 수 있습니다.

```
cat /proc/sys/net/ipv4/ip_local_port_range
```

2.2.6.5. 외부 인바운드 프록시 포트

무단 액세스로부터 프록시를 보호하려면 Uyuni 프록시에서 외부 인바운드 포트를 열어 방화벽을 구성해야 합니다.

이러한 포트를 열면 외부 네트워크 트래픽이 Uyuni 프록시에 액세스할 수 있습니다.

표 6. Uyuni 프록시의 외부 포트 요구사항

| 포트 번호 | 프로토콜 | 사용 대상 | 참고 |
|-------|---------|-------|---|
| 22 | | | 사용자가 Salt SSH로 프록시 호스트를 관리하려는 경우에만 필요합니다. |
| 67 | TCP/UDP | DHCP | 클라이언트가 서버의 IP 주소를 요청할 때만 필요합니다. |
| 69 | TCP/UDP | TFTP | 자동 클라이언트 설치를 위해 서버가 PXE로 사용될 때 필요합니다. |
| 443 | TCP | HTTPS | Web UI, 클라이언트, 서버 및 프록시(tftpsync) 요청입니다. |
| 4505 | TCP | salt | 클라이언트로부터의 통신 요청을 수락하기 위해 필요합니다. 클라이언트가 연결을 시작하면, Salt 마스터로부터 명령을 수신하기 위해 열려 있는 상태를 유지합니다. |

| 포트 번호 | 프로토콜 | 사용 대상 | 참고 |
|-------|------|-------|---|
| 4506 | TCP | salt | 클라이언트로부터의 통신 요청을 수락하기 위해 필요합니다. 클라이언트가 연결을 시작하며, Salt 마스터로 결과를 다시 보고하기 위해 열려 있는 상태를 유지합니다. |
| 8022 | | | ssh-push 및 ssh-push-tunnel 연락 방법에 필수적입니다. 프록시에 연결된 클라이언트는 서버에 체크인을 시작하고 클라이언트로 훔을 통해 이동합니다. |

2.2.6.6. 외부 아웃바운드 프록시 포트

액세스할 수 있는 프록시를 제한하려면 Uyuni 프록시에서 외부 아웃바운드 포트를 열어 방화벽을 구성해야 합니다.

이러한 포트를 열면 Uyuni 프록시로부터의 네트워크 트래픽이 외부 서비스와 통신할 수 있습니다.

표 7. Uyuni 프록시의 외부 포트 요구사항

| Port number | Protocol | Used By | Notes |
|-------------|----------|---------|--|
| 80 | | | Used to reach the server. |
| 443 | TCP | HTTPS | Required for SUSE Customer Center. |
| 4505 | TCP | Salt | Required to connect to Salt master either directly or via proxy. |
| 4506 | TCP | Salt | Required to connect to Salt master either directly or via proxy. |

2.2.6.7. 외부 클라이언트 포트

Uyuni 서버와 클라이언트 사이에서 방화벽을 구성하려면 외부 클라이언트 포트가 열려 있어야 합니다.

대부분의 경우에는 이러한 포트를 조정할 필요가 없습니다.

표 8. Uyuni 클라이언트의 외부 포트 요구사항

| Port number | Direction | Protocol | Notes |
|-------------|-----------|----------|--|
| 22 | Inbound | SSH | Required for ssh-push and ssh-push-tunnel contact methods. |
| 80 | Outbound | | Used to reach the server or proxy. |
| 443 | Outbound | | Used to reach the server or proxy. |
| 4505 | Outbound | TCP | Required to connect to Salt master either directly or via proxy. |
| 4506 | Outbound | TCP | Required to connect to Salt master either directly or via proxy. |

| Port number | Direction | Protocol | Notes |
|-------------|-----------|----------|--|
| 9090 | Outbound | TCP | Required for Prometheus user interface. |
| 9093 | Outbound | TCP | Required for Prometheus alert manager. |
| 9100 | Outbound | TCP | Required for Prometheus node exporter. |
| 9117 | Outbound | TCP | Required for Prometheus Apache exporter. |
| 9187 | Outbound | TCP | Required for Prometheus PostgreSQL. |

2.2.6.8. 필수 URL

Uyuni에서 클라이언트를 등록하고 업데이트를 수행하기 위해 액세스할 수 있어야 하는 URL이 몇 개 있습니다. 대부분의 경우에는 해당 URL에 대한 액세스를 허용하는 것으로 충분합니다.

- scc.suse.com
- updates.suse.com
- installer-updates.suse.com
- registry.suse.com
- registry-storage.suse.com
- opensuse.org

또한 SUSE 제품이 아닌 제품의 경우 다음 URL에 액세스해야 할 수도 있습니다.

- download.nvidia.com
- public.dhe.ibm.com
- nu.novell.com

You can find additional details on whitelisting the specified URLs and their associated IP addresses in this article: [Accessing SUSE Customer Center and SUSE registry behind a firewall and/or through a proxy](#).

SUSE 이외의 클라이언트를 사용하는 경우에는 해당 운영 체제용 특정 패키지를 제공하는 다른 서버에 대한 액세스도 허용해야 할 수 있습니다. 예를 들어, Ubuntu 클라이언트가 있는 경우 Ubuntu 서버에 액세스할 수 있어야 합니다.

SUSE 이외의 클라이언트에 대한 방화벽 액세스 문제 해결과 관련한 자세한 내용은 **Administration > Troubleshooting**에서 확인할 수 있습니다.

2.3. 공용 클라우드 요구사항

이 섹션에서는 공용 클라우드 인프라에 Uyuni를 설치하기 위한 요구사항을 제공합니다. Amazon EC2, Google Compute Engine, Microsoft Azure에서 이러한 지침을 테스트했지만 약간의 차이는 있지만 다른 공급자에서도 작동해야 합니다.

시작하기 전 고려해야 할 사항은 다음과 같습니다.

- Uyuni 설정 절차는 정방향 확인 된 역방향 DNS 조회를 수행합니다. 설정 절차를 완료하고 Uyuni이 예상대로 작동하려면 이 작업이 성공해야 합니다. Uyuni를 설정하기 전에 호스트 이름 및 IP 구성을 수행하는 것이 중요합니다.
- Uyuni 서버 및 프록시 인스턴스는 DNS 항목을 관리할 수 있는 네트워크 구성에서 실행해야 하지만 인터넷에서 전체적으로 액세스할 수 없습니다.
- 이 네트워크 구성에서는 DNS 확인이 제공되어야 합니다. **hostname -f**에서 FQDN(정규화된 도메인 이름)을 반환해야 합니다.
- DNS 확인은 클라이언트 연결에도 중요합니다.
- DNS는 선택한 클라우드 프레임워크와 독립적입니다. 자세한 지침은 클라우드 공급자의 설명서를 참조하십시오.
- 외부 가상 디스크에서 소프트웨어 리포지토리, 서버 데이터베이스 및 프록시 squid 캐시를 찾는 것이 좋습니다. 이를 수행하면 인스턴스가 예기치 않게 종료되는 경우 데이터 손실을 방지할 수 있습니다. 이 섹션에는 외부 가상 디스크를 설정하기 위한 지침이 포함되어 있습니다.

2.3.1. 네트워크 요구사항

공용 클라우드에서 Uyuni를 사용하는 경우 제한 네트워크를 사용해야 합니다. 방화벽이 올바르게 설정된 VPC 개인 서브넷을 사용하는 것이 좋습니다. 지정된 IP 범위의 시스템만 인스턴스에 액세스할 수 있습니다.



- 퍼블릭 클라우드에서 Uyuni를 실행하면 강력한 보안 조치를 구현할 수 있습니다. 인스턴스에 대한 액세스 제한, 필터링, 모니터링 및 감사 기능은 필수 기능입니다. SUSE는 적절한 경계 보안이 부족한 전역 액세스 Uyuni 인스턴스를 사용하지 않을 것을 강력하게 권장됩니다.

Uyuni Web UI에 액세스하려면, 네트워크 액세스 제어를 구성할 때 HTTPS를 허용하십시오. 이렇게 하면 Uyuni Web UI에 액세스할 수 있습니다.

EC2 및 Azure에서 새 보안 그룹을 만들고 HTTPS에 대한 인바운드 및 아웃바운드 규칙을 추가합니다. GCE에서 **방화벽** 섹션 아래의 **HTTPS 트래픽 허용** 상자를 선택합니다.

2.3.2. 스토리지 볼륨 준비

Uyuni용 리포지토리와 데이터베이스를 루트 볼륨과 별도의 스토리지 장치에 저장하는 것이 좋습니다. 이렇게 하면 데이터 손실을 방지하고 성능을 향상하는데 도움이 됩니다.

Uyuni 컨테이너는 기본 스토리지 위치를 사용합니다. 이러한 위치는 사용자 정의 스토리지를 배포하기 전에 구성해야 합니다. 자세한 내용은 **Installation-and-upgrade > Container-management**에서 확인할 수 있습니다.



- 공용 클라우드 설치에는 논리적 볼륨 관리(LVM)를 사용하지 않아야 합니다.

리포지토리 저장소를 위한 디스크 크기는 Uyuni로 관리할 배포 및 채널 수에 따라 다릅니다. 가상 디스크를 연결하면 인스턴스에 Unix 장치 노드로 표시됩니다. 장치 노드의 이름은 공급자 및 선택한 인스턴스 유형에 따라 다릅니다.

Uyuni 서버의 루트 볼륨이 100GB 이상인지 확인합니다. 500GB 이상의 추가 저장소 디스크를 추가하고 가능하면 SSD 저장소를 선택하십시오. Uyuni 서버의 클라우드 이미지는 인스턴스가 시작될 때 스크립트를 사용하여 별도의 볼륨을 할당합니다.

인스턴스를 시작할 때 Uyuni 서버에 로그인하고 이 명령을 사용하여 사용 가능한 모든 저장소 장치를 찾을 수 있습니다.

```
hwinfo --disk | grep -E "장치 파일:"
```

선택해야 할 장치가 확실하지 않은 경우 **lsblk** 명령을 사용하여 각 장치의 이름과 크기를 확인하십시오. 찾고 있는 가상 디스크의 크기와 일치하는 이름을 선택하십시오.

mgr-storage-server 명령어로 외부 디스크를 설정할 수 있습니다. 그러면 **/manager_storage**에 마운트된 XFS 파티션이 생성되고 이를 데이터베이스 및 리포지토리의 위치로 사용합니다.

```
/usr/bin/mgr-storage-server <devicename>
```

장 3. 배포 및 설치

3.1. Uyuni 서버 설치

Uyuni 서버를 배포하는 데는 다양한 시나리오가 있습니다.

3.1.1. Uyuni Server Deployment on openSUSE Tumbleweed

3.1.1.1. Deployment Preparations

이 섹션에서는 Uyuni 서버 설정 및 배포에 대한 전문 지식을 습득할 수 있습니다. 이 프로세스는 **Podman**, **Uyuni 컨테이너 유ти리티**의 설치, 배포, **mgrctl**을 통해 컨테이너와의 상호작용을 시작하는 프로세스로 구성됩니다.



This section assumes you have already configured an openSUSE Tumbleweed host server, whether it is running on a physical machine or within a virtual environment.

<https://download.opensuse.org/tumbleweed/>

3.1.1.2. 컨테이너 호스트 일반 요구사항

일반 요구사항은 **Installation-and-upgrade > General-requirements**에서 확인할 수 있습니다.

An openSUSE Tumbleweed server should be installed from installation media.

<https://download.opensuse.org/tumbleweed/>

이 절차는 아래에 설명되어 있습니다.

3.1.1.3. 컨테이너 호스트 요구사항

CPU, RAM, 스토리지 요구사항은 **Installation-and-upgrade > Hardware-requirements**에서 확인할 수 있습니다.



클라이언트가 FQDN 도메인 이름을 확인할 수 있도록 하려면 컨테이너화된 서버와 호스트 컴퓨터가 모두 올바르게 작동하는 DNS 서버에 연결되어 있어야 합니다. 또한 역방향 확인도 올바르게 구성해야 합니다.

3.1.1.4. Installing Uyuni Tools For Use With Containers

Procedure: Installing Uyuni Tools on openSUSE Tumbleweed

1. On your local host, open a terminal window and log in.
2. Add the following repository to your openSUSE Tumbleweed server.
You might need to use **sudo** for the following commands.

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/images/repo/Uyuni-Server-POOL-$(arch)-Media1/ uyuni-server-stable
```

3. Refresh the repository list and import the key:

```
zypper ref
```

When prompted, trust and import the new repository GPG key.

4. 컨테이너 도구를 설치합니다.

```
zypper in mgradm mgrctl mgradm-bash-completion mgrctl-bash-completion uyuni-storage-setup-server
```

Uyuni 컨테이너 유ти리티에 대한 자세한 내용은 [Uyuni 컨테이너 유ти리티](#)를 참조하십시오.

3.1.1.5. 사용자 정의 영구 스토리지 구성

이 단계는 선택 사항입니다. 그러나 인프라에 사용자 정의 영구 스토리지가 필요한 경우 **mgr-storage-server** 도구를 사용하십시오.

자세한 내용은 **mgr-storage-server --help**를 참조하십시오. 이 도구는 컨테이너 스토리지 및 데이터베이스 볼륨 생성을 간소화합니다.

다음 방법으로 명령 사용:

```
mgr-storage-server <storage-disk-device> [<database-disk-device>]
```

예:

```
mgr-storage-server /dev/nvme1n1 /dev/nvme2n1
```

이 명령은 **/var/lib/containers/storage/volumes**에 영구 스토리지 볼륨을 생성합니다.



자세한 내용은 **Installation-and-upgrade** > **Container-management**에서 확인할 수 있습니다.

3.1.1.6. Deploying an Uyuni Container With Podman

3.1.1.6.1. mgradm 개요

[command] **mgradm** 도구를 사용하여 Uyuni(를) 컨테이너로 배포합니다. Uyuni 서버는 2가지 방법으로 컨테이너로 배포할 수 있습니다. 이 섹션에서는 기본 컨테이너 배포를 중심으로 설명합니다.

사용자 정의 구성 파일을 사용하여 배포하는 방법에 대한 자세한 내용은 **Installation-and-upgrade** > **Container-management**에서 확인할 수 있습니다.

자세한 내용은 명령줄에서 **mgradm --help**를 실행하여 확인할 수 있습니다.



보안을 위해 강화된 Uyuni 서버 호스트는 **/tmp** 폴더의 파일 실행을 제한할 수 있습니다. 이러한

경우 해결 방법으로 **TMPDIR** 환경 변수를 다른 기존 경로로 내보낸 다음, **mgradm**을 실행합니다.

예:

```
export TMPDIR=/path/to/other/tmp
```

Uyuni 업데이트에서는 이 해결 방법이 불필요하도록 도구가 변경될 예정입니다.

절차: Podman을 사용한 Uyuni 컨테이너 배포

- 터미널에서 sudo 사용자 또는 루트로 다음 명령을 실행합니다.

```
sudo mgradm install podman
```

컨테이너를 sudo 또는 루트로 배포해야 합니다. 이 단계를 건너뛰면 터미널에 다음과 같은 오류가 표시됩니다.

!

```
INF Setting up uyuni network
9:58AM INF Enabling system service
9:58AM FTL Failed to open /etc/systemd/system/uyuni-server.service
for writing
error="open /etc/systemd/system/uyuni-server.service: permission
denied"
```

- 배포가 완료될 때까지 기다립니다.
- 브라우저를 열고 서버 FQDN으로 이동합니다.

3.1.1.6.2. 영구 볼륨

많은 사용자가 영구 볼륨의 위치를 지정하기를 원할 것입니다.

i If you are just testing out Uyuni you do not need to specify these volumes. **mgradm** will setup the correct volumes by default.

볼륨 위치 지정은 일반적으로 대규모 프로덕션 배포에서 사용됩니다.

기본적으로 [command] **podman**은 볼륨을 **/var/lib/containers/storage/volumes**에 저장합니다.

You can provide custom storage for the volumes by mounting disks on this path or the expected volume path inside it such as: **/var/lib/containers/storage/volumes/var-spacewalk**. This is especially important for the database and package mirrors.

For a list of all persistent volumes in the container, see:

- Installation-and-upgrade > Container-management
- Administration > Troubleshooting

3.1.2. Uyuni 서버 air-gapped 배포

3.1.2.1. air-gapped 배포란?

air-gapped 배포란 안전하지 않은 네트워크, 특히 인터넷으로부터 물리적으로 격리된 네트워크 시스템을 설정하고 운영하는 것을 의미합니다. 이러한 유형의 배포는 일반적으로 군사 설치, 금융 시스템, 중요 인프라 및 민감한 데이터를 처리하고 외부 위협으로부터 보호해야 하는 모든 곳에서 보안 수준이 높은 환경에서 사용됩니다.

You can easily pull container images using **Podman** or **Docker** on a machine with internet access.

Procedure: Pulling the images

- Pull the desired images, then save the images as a **tar** archive. For example:

목록 1. Podman

```
podman pull registry.opensuse.org/uyuni/server:latest
registry.opensuse.org/uyuni/server-postgresql:latest
podman save --output images.tar registry.opensuse.org/uyuni/server:latest
registry.opensuse.org/uyuni/server-postgresql:latest
```

목록 2. Docker

```
docker pull registry.opensuse.org/uyuni/server:latest
registry.opensuse.org/uyuni/server-postgresql:latest
docker save --output images.tar registry.opensuse.org/uyuni/server:latest
registry.opensuse.org/uyuni/server-postgresql:latest
```

- Transfer the resulting **images.tar** to the Server container host and load it using the following command:

목록 3. Load the server image

```
podman load -i images.tar
```

3.1.2.2. Obtaining container images for Salt formulas in air-gapped environments

Some formulas, like Bind and DHCP (Kea), also use containers. If you plan to use them in an air-gapped environment, you need to pull their images, save them to an archive, and load them on your Uyuni Proxy or another managed system.

The images are available from registry.opensuse.org.

Procedure: Obtaining formula images for air-gapped environments

- On a system with Internet access, pull the required images. For example:

```
podman pull registry.opensuse.org/opensuse/bind:latest
podman pull registry.opensuse.org/opensuse/kea:latest
```

- Save the images to a TAR archive:

```
podman save -o formula-images.tar registry.opensuse.org/opensuse/bind:latest
registry.opensuse.org/opensuse/kea:latest
```

3. Transfer the **formula-images.tar** file to your air-gapped system.
4. Load the images on the air-gapped system:

```
podman load -i formula-images.tar
```

3.1.2.2.1. Deploy Uyuni on openSUSE Tumbleweed

또한 Uyuni은(는) 필요한 모든 컨테이너 이미지를 시스템에 설치할 수 있는 RPM 패키지로 제공합니다.



- 사용자는 필요한 RPM을 내부 네트워크에서 사용할 수 있도록 해야 합니다. 이 작업 두 번째 Uyuni 서버 또는 일종의 미러를 사용하여 수행할 수 있습니다.

Procedure: Install Uyuni on openSUSE Tumbleweed in air-gapped environment

1. Install openSUSE Tumbleweed.
2. 시스템을 업데이트합니다.
3. 도구 패키지 및 이미지 패키지 설치(\$ARCH\$를 올바른 아키텍처로 대체):

```
zypper install mgradm* mgrctl* uyuni-server*-image*
```

4. Uyuni을(를) **mgradm**으로 배포합니다. Air-gapped 환경에서는 **--pullPolicy Never** 옵션을 사용할 수 있습니다.

For more detailed information about installing Uyuni Server on openSUSE Tumbleweed, see [Server Deployment](#).

Uyuni 서버를 업그레이드하려면 시스템의 모든 패키지를 업그레이드하고 [서버 업그레이드](#)에 정의된 절차를 따릅니다.

3.2. Install Uyuni Proxy

There are various scenarios to deploy a Uyuni Proxy. All these scenarios presume you have already successfully deployed a Uyuni 2026.01 Server.

3.2.1. 컨테이너화된 Uyuni 프록시 설정

Uyuni 프록시 컨테이너에 대한 컨테이너 호스트가 준비되면 컨테이너 설정에서 구성을 완료하기 위해 몇 가지 추가 단계가 필요합니다.

Procedure

1. Uyuni 프록시 구성 아카이브 파일 생성
2. 설치 단계에서 준비한 컨테이너 호스트로 구성 아카이브를 전송하고 압축을 풉니다.

3. **mgrpxy**를 실행하여 프록시 서비스를 시작합니다.

3.2.1.1. 프록시 구성 생성

Uyuni 프록시의 구성 아카이브는 Uyuni 서버에서 생성됩니다. 각 추가 프록시에는 자체 구성 아카이브가 필요합니다.

컨테이너화된 Uyuni 프록시의 경우 변경 사항을 적용하려면 새 프록시 구성 파일을 빌드한 다음 컨테이너를 다시 배포해야 합니다. 이는 SSL 인증서를 포함한 설정을 업데이트하는 프로세스입니다.



Podman 배포의 경우, 이 프록시 구성을 생성하기 전에 Uyuni 프록시의 컨테이너 호스트가 Uyuni 서버에 클라이언트로 등록되어 있어야 합니다.

프록시 FQDN을 사용하여 등록된 클라이언트가 아닌 프록시 컨테이너 구성을 생성하는 경우(Kubernetes 사용 사례와 같이), 시스템 목록에 새 시스템 항목이 표시됩니다. 이 새 항목은 이전에 입력한 프록시 FQDN 값 아래에 표시되며 시스템 유형은 **외부**입니다.

Peripheral servers are always using third-party SSL certificates. If the hub server has generated the certificates for the peripheral server, it needs to generate the certificate of each proxy too.

On the hub server, run the following command.



```
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=PROXY --set-cname="proxy.example.com"
```

The files to use will be

1. **/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT** as the root CA,
2. **/root/ssl-build/<hostname>/server.crt** as the proxy certificate and
3. **/root/ssl-build/<hostname>/server.key** as the proxy certificate's key.

3.2.1.1.1. Web UI를 사용하여 프록시 구성 생성

Procedure: Generating a Proxy Container Configuration Using Web UI

1. Web UI에서 **Systems** > **프록시 구성**으로 이동하여 필요한 데이터를 입력합니다.
2. **Proxy FQDN** 필드에 프록시의 정규화된 도메인 이름을 입력합니다.
3. 상위 **FQDN** 필드에 Uyuni Server 또는 다른 Uyuni Proxy에 대한 정규화된 도메인 이름을 입력하십시오.
4. **프록시 SSH 포트** 필드에 SSH 서비스가 Uyuni Proxy에서 수신 대기하는 SSH 포트를 입력하십시오. 권장 사항은 기본 포트인 8022를 유지하는 것입니다.
5. In the **Max Squid cache size [MB]** field type maximal allowed size for Squid

cache. Recommended is to use at most 80% of available storage for the containers.



2GB는 기본 프록시 squid 캐시 크기입니다. 사용자의 환경에 적합하도록 조정해야 합니다.

SSL 인증서 선택 목록에서 Uyuni 프록시에 대해 새 서버 인증서를 생성해야 하는지 또는 기존 인증서를 사용해야 하는지 선택합니다. 생성된 인증서를 Uyuni 기본 제공(자체 서명) 인증서로 간주할 수 있습니다.

선택에 따라 새 인증서를 생성하기 위해 CA 인증서에 서명할 경로 또는 프록시 인증서로 사용할 기존 인증서 및 해당 키에 대한 경로를 입력하십시오.

서버에서 생성된 CA 인증서는 `/var/lib/containers/storage/volumes/root/_data/ssl-build` 디렉토리에 저장됩니다.

기존 또는 사용자 정의 인증서와 기업 및 중간 인증서의 개념에 대한 자세한 내용은 **Administration > Ssl-certs-imported**에서 확인할 수 있습니다.

6. [생성]을 클릭하여 Uyuni 서버에 새 프록시 FQDN을 등록하고 컨테이너 호스트에 대한 세부사항이 포함된 구성 아카이브(**config.tar.gz**)를 생성합니다.
7. 잠시 후 다운로드할 파일이 표시됩니다. 이 파일을 로컬에 저장합니다.

3.2.1.2. spacecmd 및 자체 서명 인증서를 사용하여 프록시 구성 생성

You can generate a Proxy configuration using **spacecmd**.

절차: spacecmd 및 자체 서명 인증서를 사용하여 프록시 구성 생성

1. 컨테이너 호스트에 SSH로 연결합니다.
2. 서버 및 프록시 FQDN을 바꾸는 다음 명령을 실행합니다.

```
mgrctl exec -ti 'spacecmd proxy_container_config_generate_cert -- dev-  
pxy.example.com dev-srv.example.com 2048 email@example.com -o  
/tmp/config.tar.gz'
```

3. 서버 컨테이너에서 생성된 구성을 복사합니다.

```
mgrctl cp server:/tmp/config.tar.gz .
```

3.2.1.1.3. spacecmd 및 사용자 정의 인증서를 사용하여 프록시 구성 생성

spacecmd를 사용하여 기본 자체 서명 인증서가 아닌 사용자 정의 인증서에 대해 프록시 구성 생성할 수 있습니다.

절차: spacecmd 및 사용자 정의 인증서를 사용하여 프록시 구성 생성

1. 서버 컨테이너 호스트에 SSH로 연결합니다.
2. Execute the following commands, replacing the Server and Proxy FQDN:

```
for f in ca.crt proxy.crt proxy.key; do
    mgrctl cp $f server:/tmp/$f
done
mgrctl exec -ti 'spacecmd proxy_container_config -- -p 8022 pxy.example.com
srv.example.com 2048 email@example.com /tmp/ca.crt /tmp/proxy.crt
/tmp/proxy.key -o /tmp/config.tar.gz'
```

3. 설정에서 중간 CA를 사용하는 경우, 이를 복사하여 **-i** 옵션과 함께 명령에 포함시킵니다(필요한 경우 여러 번 제공 가능).

```
mgrctl cp intermediateCA.pem server:/tmp/intermediateCA.pem
mgrctl exec -ti 'spacecmd proxy_container_config -- -p 8022 -i
/tmp/intermediateCA.pem pxy.example.com srv.example.com 2048 email@example.com
/tmp/ca.crt /tmp/proxy.crt /tmp/proxy.key -o /tmp/config.tar.gz'
```

4. 서버 컨테이너에서 생성된 구성을 복사합니다.

```
mgrctl cp server:/tmp/config.tar.gz .
```

3.2.1.2. Transfer Uyuni Proxy Configuration

spacecmd 명령과 Web UI 방식을 통한 생성 모두 구성 아카이브를 생성합니다. 이 아카이브는 컨테이너 호스트에서 사용할 수 있도록 해야 합니다. 생성된 아카이브를 컨테이너 호스트로 전송합니다.

3.2.1.3. Start Uyuni Proxy Containers

컨테이너는 **mgrpxy** 명령으로 시작할 수 있습니다.

Procedure: Start Uyuni Proxy Containers

1. Run command:

```
mgrpxy start uyuni-proxy-pod
```

2. Check if all containers started up as expected by calling:

```
podman ps
```

5개의 Uyuni 프록시 컨테이너가 존재해야 하며, **proxy-pod** 컨테이너 포드의 일부여야 합니다.

- proxy-salt-broker
- proxy-httpd
- proxy-tftpd
- proxy-squid
- proxy-ssh

3.2.2. Uyuni Proxy Deployment on openSUSE Tumbleweed

이 가이드에서는 Uyuni 2026.01 프록시에 대한 배포 프로세스를 간략하게 설명합니다. 이 가이드에서는 Uyuni 2026.01 서버가 이미 배포된 상태를 가정합니다. 배포하려면 다음 작업을 수행합니다.

검사 목록: 프록시 배포

1. 하드웨어 요구사항을 검토합니다.
2. Install openSUSE Tumbleweed on a bare-metal machine.
3. 프록시를 Salt 미니언으로 부트스트랩합니다.
4. 프록시 구성 파일을 생성합니다.
5. 프록시 구성 파일을 서버에서 프록시로 전송합니다.
6. 프록시 구성 파일을 사용하여 Salt 미니언을 Uyuni의 프록시로 등록합니다.

프록시 컨테이너 호스트에 지원되는 운영 체제

The supported operating system for the container host is openSUSE Tumbleweed.

컨테이너 호스트



컨테이너 호스트는 컨테이너를 관리하고 배포할 수 있는 Podman과 같은 컨테이너 엔진이 탑재된 서버입니다. 이러한 컨테이너는 애플리케이션과 라이브러리와 같은 필수적인 부분을 보관하지만, 전체 운영 체제는 보관하지 않으므로 경량화됩니다. 이 설정을 통해 애플리케이션이 다양한 환경에서 동일한 방식으로 실행될 수 있습니다. 컨테이너 호스트는 이러한 컨테이너에 CPU, 메모리, 스토리지 등 필요한 리소스를 제공합니다.

3.2.2.1. 프록시의 하드웨어 요구사항

이 테이블은 Uyuni 프록시를 배포하기 위한 하드웨어 요구사항을 보여줍니다.

표 9. 프록시 하드웨어 요구사항

| 하드웨어 | 세부 정보 | 권장 |
|------|-------------|-----------------------|
| CPU | x86-64, ARM | 최소 2개의 전용 64비트 CPU 코어 |
| RAM | 최소 | 2GB |
| | 권장 | 8GB |

| 하드웨어 | 세부 정보 | 권장 |
|--------|---|---|
| 디스크 공간 | / (루트 디렉토리) | 최소 40GB |
| | /var/lib/containers/storage/volumes/srv-www | 최소 100GB, 스토리지 요구사항은 사용할 ISO 배포 이미지, 컨테이너 및 부트스트랩 리포지토리 수에 따라 계산해야 합니다. |
| | /var/lib/containers/storage/volumes | 최소 100GB, 사용 예정인 ISO 배포 이미지, 컨테이너 및 부트스트랩 리포지토리의 수에 따라 저장 공간 요구 사항을 계산해야 합니다. |

3.2.2.2. 컨테이너 호스트 일반 요구사항

일반 요구사항은 [Installation-and-upgrade > General-requirements](#)에서 확인할 수 있습니다.

An openSUSE Tumbleweed server should be installed from installation media. This procedure is described below.

3.2.2.3. 컨테이너 호스트 요구사항

CPU, RAM, 스토리지 요구사항은 [Installation-and-upgrade > Hardware-requirements](#)에서 확인할 수 있습니다.



- 클라이언트가 FQDN 도메인 이름을 확인할 수 있도록 하려면 컨테이너화된 서버와 호스트 컴퓨터가 모두 올바르게 작동하는 DNS 서버에 연결되어 있어야 합니다. 또한 역방향 확인도 올바르게 구성해야 합니다.

3.2.2.4. Installing Uyuni Tools for Use With Containers

Procedure: Installing Uyuni Tools on openSUSE Tumbleweed

1. On your local host open a terminal window or start up a virtual machine running openSUSE Tumbleweed.
2. 로그인합니다.
3. Add the following repository to your openSUSE Tumbleweed server:

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/images/repo/Uyuni-Proxy-POOL-$(arch)-Media1/ uyuni-proxy-stable
```

4. 리포지토리 목록을 새로 고치고 키를 수락합니다.

```
zypper ref
```

5. 컨테이너 도구를 설치합니다.

```
zypper in mgrpxy mgrpxy-bash-completion uyuni-storage-setup-proxy
```



- 또는 **mgrpxy-zsh-completion** 또는 **mgrpxy-fish-completion**을 설치할 수 있습니다.

Uyuni 컨테이너 유틸리티에 대한 자세한 내용은 [Uyuni 컨테이너 유틸리티](#)를 참조하십시오.

3.2.2.5. 사용자 정의 영구 스토리지 구성

이 단계는 선택 사항입니다. 그러나 인프라에 사용자 정의 영구 스토리지가 필요한 경우 **mgr-storage-proxy** 도구를 사용하십시오.

자세한 내용은 **mgr-storage-proxy --help**를 참조하십시오. 이 도구는 컨테이너 스토리지 및 Squid 캐시 볼륨 생성을 간소화합니다.

다음 방법으로 명령 사용:

```
mgr-storage-proxy <storage-disk-device>
```

예:

```
mgr-storage-proxy /dev/nvme1n1
```



이 명령은 **/var/lib/containers/storage/volumes**에 영구 스토리지 볼륨을 생성합니다.

자세한 내용은 다음을 참조하십시오.

- Installation-and-upgrade > Container-management
- Administration > Troubleshooting

3.2.2.6. 프록시 호스트를 미니언으로 부트스트랩

작업: 프록시 호스트 부트스트랩

1. 시스템 > 부트스트랩을 선택합니다.
2. 프록시 호스트의 필드를 입력합니다.
3. 드롭다운에서 이전 단계에서 생성한 활성화 키를 선택합니다.
4. [+ 부트스트랩]을 클릭합니다.
5. 부트스트랩 프로세스가 완료될 때까지 기다립니다. Salt 메뉴를 선택한 후 Salt 미니언 키가 나열되고 수락되었는지 확인합니다.
6. 프록시 호스트를 재부팅합니다.

7. 시스템 목록에서 호스트를 선택하고 모든 이벤트가 완료된 후 두 번째 재부팅을 트리거하여 온보딩을 완료합니다.

작업: 프록시 호스트 업데이트

1. 시스템 목록에서 호스트를 선택하고 모든 패치를 적용하여 업데이트합니다.
2. 프록시 호스트를 재부팅합니다.

3.2.2.7. 프록시 구성 생성

Uyuni 프록시의 구성 아카이브는 Uyuni 서버에서 생성됩니다. 각 추가 프록시에는 자체 구성 아카이브가 필요합니다.



- Uyuni 프록시의 컨테이너 호스트는 이 프록시 구성 생성하기 전에 Uyuni 서버에 salt 미니언으로 등록해야 합니다.

다음 작업을 수행합니다.

절차

1. 프록시 구성 파일을 생성합니다.
2. 구성을 프록시로 전송합니다.
3. [literal] **mgrpxy** 명령어로 프록시를 시작합니다.

작업: 웹 UI를 사용하여 프록시 컨테이너 구성 생성

1. Web UI에서 **Systems** > **프록시 구성**으로 이동하여 필요한 데이터를 입력합니다.
2. **Proxy FQDN** 필드에 프록시의 정규화된 도메인 이름을 입력합니다.
3. **상위 FQDN** 필드에 Uyuni Server 또는 다른 Uyuni Proxy에 대한 정규화된 도메인 이름을 입력하십시오.
4. **프록시 SSH 포트** 필드에 SSH 서비스가 Uyuni Proxy에서 수신 대기하는 SSH 포트를 입력하십시오. 권장 사항은 기본 포트인 8022를 유지하는 것입니다.
5. **최대 Squid 캐시 크기 [MB]** 필드에 Squid 캐시에 허용되는 최대 크기를 입력하십시오. 일반적으로 이는 컨테이너에 사용할 수 있는 저장소의 최대 60%여야 합니다. **SSL 인증서** 선택 목록에서 Uyuni 프록시에 대해 새 서버 인증서를 생성해야 하는지 또는 기존 인증서를 사용해야 하는지 선택합니다. 생성된 인증서를 Uyuni 기본 제공(자체 서명) 인증서로 간주할 수 있습니다.

선택에 따라 새 인증서를 생성하기 위해 CA 인증서에 서명할 경로 또는 프록시 인증서로 사용할 기존 인증서 및 해당 키에 대한 경로를 입력하십시오.

The CA certificates generated on the server are stored in the `/var/lib/containers/storage/volumes/root/_data/ssl-build` directory.

기존 또는 사용자 정의 인증서와 기업 및 중간 인증서의 개념에 대한 자세한 내용은 **Administration** > **Ssl-certs-imported**에서 확인할 수 있습니다.

6. [생성]을 클릭하여 Uyuni Server에 새 프록시 FQDN을 등록하고 컨테이너 호스트에 대한 세부 정보가 포함된 구성 아카이브를 생성하십시오.
7. 잠시 후 다운로드할 파일이 표시됩니다. 이 파일을 로컬에 저장하십시오.

3.2.2.8. 프록시 구성 전송

Web UI는 구성 아카이브를 생성합니다. 이 아카이브는 프록시 컨테이너 호스트에서 사용할 수 있도록 설정해야 합니다.

작업: 프록시 구성 복사

- Copy the files to the Proxy host:

```
scp config.tar.gz <proxy-FQDN>:/root
```

- 다음을 사용하여 프록시 설치:

```
mgrpxy install podman config.tar.gz
```

3.2.2.9. Uyuni 2026.01 프록시 시작

이제 **mgrpxy** 명령으로 컨테이너를 시작할 수 있습니다.

작업: 프록시 시작 및 상태 확인

- 다음을 호출하여 프록시 시작:

```
mgrpxy start
```

- 다음을 호출하여 컨테이너 상태 확인:

```
mgrpxy status
```

Five Uyuni Proxy containers should be present and should be part of the **proxy-pod** container pod:

- proxy-salt-broker
- proxy-httpd
- proxy-tftpd
- proxy-squid
- proxy-ssh

3.2.2.9.1. 서비스에 사용자 정의 컨테이너 이미지 사용

기본적으로 Uyuni 프록시 제품군은 각 서비스에 대해 동일한 이미지 버전과 레지스트리 경로를 사용하도록 설정되어 있습니다. 그러나 **-tag** 및 **-image**로 끝나는 설치 파라미터를 사용하여 특정 서비스에 대한 기본값을 재정의할 수 있습니다.

예를 들어, 다음과 같이 사용합니다.

```
mgrpxy install podman --httpd-tag 0.1.0 --httpd-image registry.opensuse.org/uyuni/proxy-
httpd /path/to/config.tar.gz
```

이는 다시 시작하기 전에 `registry.opensuse.org/uyuni/proxy-httpsd`가 사용할 이미지이고 **0.1.0**이 버전 태그인 httpd 서비스의 구성 파일을 조정합니다.

값을 기본값으로 재설정하려면 해당 파라미터 없이 설치 명령을 다시 실행합니다.

```
mgrpxy install podman /path/to/config.tar.gz
```

이 명령은 먼저 모든 서비스의構성을 전역 기본값으로 재설정한 다음 다시 로드합니다.

3.2.3. 클라이언트에서의 프록시 변환

3.2.3.1. Overview

이 장에서는 Web UI를 사용하여 클라이언트 시스템을 Uyuni 프록시로 변환하는 방법에 대해 설명합니다.

프록시 호스트 시스템이 이미 부트스트랩되어 기본 운영 체제 채널을 구독하고 있다고 가정합니다.

클라이언트 온보딩에 대한 자세한 내용은 [Client-configuration](#) > [Registration-overview](#)에서 확인할 수 있습니다.

3.2.3.2. 요구사항

변환을 시작하기 전에 다음 요구 사항이 충족되었는지 확인하십시오.

3.2.3.2.1. 클라이언트는 다음과 같아야 함

- Uyuni에 이미 온보딩됨
- 네트워크를 통해 연결 가능

3.2.3.3. Preparation

프록시 변환을 계속하기 전에 변환 과정 중 중단을 방지하기 위해 다음 준비 사항이 완료되었는지 확인하십시오.

3.2.3.3.1. SSL Certificates

프록시와 다른 구성 요소 간의 통신을 보호하려면 유효한 SSL 인증서가 필요합니다.

필요 사항:

- Uyuni 서버의 인증서에 서명한 인증 기관(CA)의 공개 인증서
- 프록시용 인증서.
- 프록시 인증서에 해당하는 개인 키입니다.



인증 기관(CA)이 임시 인증서 체인을 사용하는 경우, 모든 임시 인증서도 반드시 포함해야 합니다.

타사 인증서를 사용하지 않는 경우, Uyuni 컨테이너 내부의 `rhn-ssl-tool`을 사용하여 생성할 수 있습니다.

프록시 인증서 생성

- Uyuni 서버 호스트에서 다음을 실행:

```
mgrctl exec -ti -- rhn-ssl-tool --gen-server \
--set-hostname=<PROXY-FQDN> \
--dir="/root/ssl-build"
```

다른 파라미터에 대한 자세한 내용은 **Administration > Ssl-certs-selfsigned**에서 확인할 수 있습니다.

- 인증서를 Uyuni 서버 호스트 이전

```
mgrctl cp server:/root/ssl-build/<PROXY-FQDN>/server.crt /root/proxycert.pem
mgrctl cp server:/root/ssl-build/<PROXY-FQDN>/server.key /root/proxykey.pem
mgrctl cp server:/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT /root/rootca.pem
```

 인증서 및 키 파일이 생성된 정확한 폴더를 확인하려면 다음 명령으로 디렉토리를 나열할 수 있습니다.

```
mgrctl exec -ti -- ls -ltd /root/ssl-build/*/
```

- Uyuni 서버 호스트에서 인증서 이전

```
scp <UYUNI-FQDN>:/root/proxycert.pem ./
scp <UYUNI-FQDN>:/root/proxykey.pem ./
scp <UYUNI-FQDN>:/root/rootca.pem ./
```

3.2.3.3.2. 패키지 준비

Install mgrpxy

mgrpxy 도구는 시스템에 맞는 리포지토리에서 설치해야 합니다. 다음 중 적절한 리포지토리를 선택합니다.

<https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable:/ContainerUtils/>

목록 4. Example openSUSE Tumbleweed installation:

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable:/ContainerUtil
s/openSUSE_Tumbleweed/ uyuni-containerutils
zypper ref
zypper in mgrpxy
```

컨테이너 이미지 설치

컨테이너 이미지를 RPM 패키지로 배포하는 것이 권장됩니다. 클라이언트에 다음 패키지가 설치되어 있는지 확인하십시오.

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/containerfile/
uyuni-proxy-images
```

```
zypper ref
zypper in uyuni-proxy-httpd-image \
    uyuni-proxy-salt-broker-image \
    uyuni-proxy-squid-image \
    uyuni-proxy-ssh-image \
    uyuni-proxy-tftpd-image
```

air-gapped 배포에 대한 자세한 내용은 **Installation-and-upgrade > Container-deployment**에서 확인할 수 있습니다.

3.2.3.4. 프록시 클라이언트 설정

1. 클라이언트의 **Overview** 페이지로 이동합니다.
2. [프록시로 변환]을 클릭합니다.

프록시 설정 양식으로 리디렉션되었는지 확인합니다.

이 페이지는 나중에 **Details > Proxy > Configuration** 탭에서 액세스할 수 있습니다.

3. Web UI에서 **프록시 > 구성**으로 이동하여 필요한 데이터를 입력합니다.

절차: 프록시 구성

- a. 상위 **FQDN** 필드에 상위 서버 또는 프록시의 완전히 정규화된 도메인 이름을 입력합니다.
- b. **프록시 SSH 포트** 필드에 SSH 서비스가 Uyuni Proxy에서 수신 대기하는 SSH 포트를 입력하십시오. 권장 사항은 기본 포트인 8022를 유지하는 것입니다.
- c. **최대 Squid 캐시 크기** 필드에 Squid 캐시에 허용되는 최대 크기를 입력합니다.
- d. **프록시 관리자 이메일** 필드에 관리자 이메일 주소를 입력합니다.
- e. **Certificates** 섹션에서, 준비 단계에서 획득한 Uyuni 프록시용 인증서를 제공하십시오.
- f. **Source** 섹션에서 **RPM** 또는 **Registry** 옵션 중 하나를 선택합니다.
 - **RPM** 옵션은 air-gapped 또는 제한 환경에서 권장됩니다. The **Registry** option can be used if connectivity to the container image registry is available. + If selected, you will be prompted to choose between two sub-options: **Simple** or **Advanced**.
 - **Simple**을 선택한 경우, **Registry URL** 및 **Containers Tag** 필드에 값을 입력합니다.
 - **Registry URL**의 경우 `registry.opensuse.org/uyuni`를 사용합니다.
 - 드롭다운 목록에서 태그를 선택합니다.
 - **Advanced**를 선택하면 양식에 추가 섹션이 표시됩니다.
 - 각 개별 컨테이너 URL 필드에 레지스트리 [literal] `registry.opensuse.org/uyuni` 뒤에 해당 접미사(예: `proxy-httd` 또는 `salt-broker`)를 사용합니다.
 - 드롭다운 목록에서 태그를 선택합니다.

4. 모든 필드에 입력한 후에는 [적용]을 클릭하여 구성을 적용하고 프록시 설치 작업을 예약합니다.

3.2.3.5. 프록시 활성화 확인

클라이언트의 이벤트 기록을 확인하여 작업의 성공 여부를 확인합니다.

(선택 사항) 프록시의 HTTP 엔드포인트에 액세스하여 환영 페이지가 표시되는지 확인합니다.

3.2.4. K3s에 Uyuni 프록시 배포

3.2.4.1. Installing K3s

On the container host machine, install **K3s** (replace <K3S_HOST_FQDN> with the FQDN of your K3s host):

```
curl -sfL https://get.k3s.io | INSTALL_K3S_EXEC="--tls-san=<K3S_HOST_FQDN>" sh -
```

3.2.4.2. Installing Tools

설치하려면 **mgrpxy** 및 **helm** 패키지가 필요합니다.

설치 스크립트를 사용하여 Helm을 설치합니다.

```
curl -fsSL -o get_helm.sh https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3
chmod 700 get_helm.sh
./get_helm.sh
```

자세한 내용은 <https://helm.sh/docs/intro/install/#from-script>에서 확인할 수 있습니다.

The **mgrpxy** package is available in the container utils repository. Pick the one matching the distribution in: <https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable:/ContainerUtils/>.

Procedure

- Leap Micro에 패키지를 설치하려면 다음을 실행합니다.

```
transactional-update pkg install mgrpxy
```

- 재부팅합니다.

3.2.4.3. Deploying the Uyuni Proxy Helm Chart

To configure the storage of the volumes to be used by the Uyuni Proxy pod, define persistent volumes for the following claims. If you do not customize the storage configuration, K3s will automatically create the storage volumes for you.

영구 볼륨 클레임의 이름은 다음과 같습니다.

- **squid-cache-pv-claim**
- **/package-cache-pv-claim**

- /tftp-boot-pv-claim

Installation-and-upgrade → **Container-deployment**에서의 설명과 같이 Uyuni 프록시에 대한 구성을 생성합니다. 구성 **tar.gz** 파일을 복사한 다음 설치합니다.

```
mgrpxy install kubernetes /path/to/config.tar.gz
```

For more information see:

- <https://kubernetes.io/docs/concepts/storage/persistent-volumes/> (Kubernetes)
- <https://rancher.com/docs/k3s/latest/en/storage/> (K3s) 설명서

장 4. 업그레이드 및 마이그레이션

4.1. 서버

4.1.1. Migrating the Uyuni Server to openSUSE Tumbleweed

This page describes a simple, backup-and-restore migration of a Uyuni Server running on openSUSE Leap Micro 5.5 to a fresh host running openSUSE Tumbleweed as the base OS.

4.1.1.1. Overview of the Migration Process

You will:

- Create a full server backup with **mgradm backup** on the openSUSE Leap Micro 5.5 host.
- Reinstall the host with openSUSE Tumbleweed (server profile).
- Install Uyuni tools and prerequisites on Tumbleweed.
- Restore the backup with **mgradm backup restore**.
- Start services and verify the server.

4.1.1.2. Requirements and Considerations

- Source server: openSUSE Leap Micro 5.5 running Uyuni (for example: 2026.01).
- Target server: openSUSE Tumbleweed with the same hostname/FQDN and IP (recommended) to avoid client-side changes.
- SSH/scp access between machines for transferring the backup tarball.
- Sufficient free disk space on both source and target for the backup and restore.



Restore to the same Uyuni version you backed up, or a version explicitly documented as compatible for restore. If you use development or preview repositories (for example, Uyuni Master), expect changes and re-validate.

4.1.1.3. Migration Procedure

4.1.1.3.1. Step 1: Create a Backup on the openSUSE Leap Micro 5.5 Server

Procedure: Create a Backup

1. As root on the old server, create a backup directory and run the backup:

```
mgradm backup /tmp/uyuni-backup
```

2. Package the backup for transfer:

```
tar -C /tmp -cvf /tmp/uyuni-backup.tar uyuni-backup
```

3. Copy the backup to a safe location you can reach from the new host:

```
scp /tmp/uyuni-backup.tar <USER>@<HOST>:/path/to/store/
```



- You can store the backup to external storage or an object store as long as you can fetch it on the new host.

4.1.1.3.2. Step 2: Reinstall the Host with openSUSE Tumbleweed

Procedure: Reinstalling the Host

1. Reprovision the VM or bare-metal host with openSUSE Tumbleweed.
2. Choose a basic “server profile” installation.
3. Set the same hostname/FQDN and IP address as the original server if you want clients to reconnect seamlessly.

4.1.1.3.3. Step 3: Install Uyuni Tools and Prerequisites on Tumbleweed

Procedure: Installing Tools and Prerequisites

1. Add the Uyuni Stable repository and install tools:

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/images/repo/Uyuni-Server-POOL-x86_64-Media1 uyuni-server-stable
zypper ref
zypper in mgradm mgrctl mgradm-bash-completion mgrctl-bash-completion uyuni-storage-setup-server
```

2. Install Podman if it was not automatically pulled in:

```
zypper in podman
```



- The package **uyuni-storage-setup-server** provides the **mgr-storage-server** tool for preparing persistent volumes. Installing **podman** explicitly may be necessary on some installations.

4.1.1.3.4. Step 4: Optional - Prepare Persistent Storage

Procedure: Preparing Persistent Storage

It is recommended to configure persistent storage with **mgr-storage-server** to avoid container full-disk issues.

```
mgr-storage-server <storage-disk-device> [<database-disk-device>]
```

Devices must be raw (no existing filesystem). The tool creates volumes at `/var/lib/containers/storage/volumes`.



For details, see:

- [Installation-and-upgrade > Container-management](#)
- [Administration > Troubleshooting](#)

4.1.1.3.5. Step 5 Fetch and Restore the Backup on Tumbleweed

Procedure: Fetching and Restoring the Backup

1. Copy the backup to the new server and unpack it:

```
scp <USER>@<HOST>:/path/to/store/uyuni-backup.tar /tmp/
tar -C /tmp -xvf /tmp/uyuni-backup.tar
```

2. Restore using `mgradm` (point to the extracted backup directory):

```
mgradm backup restore /tmp/uyuni-backup
```

4.1.1.3.6. Step 6: Start Services and Verify

Procedure: Starting Services and Verifying

1. Start the server services:

```
mgradm start
```

2. Verify:

- Check that all containers are up: `mgrctl ps` or `podman ps`.
- Access the Web UI (HTTPS) and log in.
- Review logs for errors: `mgrctl logs server` and other components as needed.
- —

4.1.1.4. Notes and Troubleshooting

- If Podman wasn't installed automatically, install it with `zypper in podman` and rerun the restore/start steps.
- Ensure the target host has the same time, hostname, and IP configuration expected by your setup (especially if clients exist).

- For large environments, ensure adequate disk throughput and space. The backup and restore can take a long time.



If the restore fails or the new system cannot start, you can still boot the original openSUSE Leap Micro 5.5 system and continue service. Keep the original VM/snapshots until you fully validate the new Tumbleweed-based server.

4.1.2. 레거시 Uyuni 서버를 컨테이너로 마이그레이션

레거시 Uyuni 서버를 컨테이너로 마이그레이션하려면 새 시스템이 필요합니다.

이 마이그레이션의 맥락에서 레거시 Uyuni 서버(RPM 설치)를 _old server_라고도 합니다.

4.1.2.1. Requirements and Considerations

4.1.2.1.1. Hostnames

인플레이스 마이그레이션은 불가능할 뿐만 아니라, 현재 마이그레이션 절차는 호스트 이름 변경 기능을 허용하지 않습니다.

따라서 새 서버의 FQDN(정규화된 도메인 이름)은 레거시 서버의 FQDN과 동일하게 유지됩니다.



マイグレーティョン後에는 새 서버를 가리키도록 DHCP 및 DNS 레코드를 업데이트해야 합니다.

자세한 내용은 [마이그레이션 마무리](#)에서 확인할 수 있습니다.

4.1.2.1.2. SSL certificates

SSL 인증서는 후속 단계에서 필요합니다. 자체 서명 생성 CA 및 인증서를 사용하지 않는 경우 시작 전에 다음을 준비하십시오.

- 인증 기관(CA) SSL 공개 인증서. CA 체인을 사용하는 경우 모든 중간 CA도 반드시 사용 가능해야 합니다.
- 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**가 필요하며, 보고서 데이터베이스에 액세스하는 데 사용되는 FQDN이 **Subject Alternative Name**으로 지정되어야 합니다.

During a migration, the server SSL certificate and CA chain are copied from the source server, meaning that only the database certificates are required

4.1.2.2. GPG 키

- Self trusted GPG keys are not migrated.
- GPG keys that are trusted in the RPM database only are not migrated. Thus synchronizing channels with **spacewalk-repo-sync** can fail.
- 관리자는 실제 서버 마이그레이션을 수행한 후 이러한 키를 레거시 Uyuni 설치에서 컨테이너 호스트로 수동으로 마이그레이션해야 합니다.

절차: 새 서버로 GPG 키를 수동으로 마이그레이션

1. 레거시 Uyuni 서버의 키를 새 서버의 컨테이너 호스트로 복사합니다.
2. 그 후, **mgradm gpg add <PATH_TO_KEY_FILE>** 명령을 사용하여 마이그레이션된 서버에 각 키를 추가합니다.

4.1.2.2.1. 레거시 서버에서의 초기 준비

복제해야 할 데이터의 양에 따라 마이그레이션은 매우 오랜 시간이 소요될 수 있습니다. 다운타임을 줄이기 위해, 레거시 서버의 모든 서비스가 가동된 상태를 유지하면서 initial replication, re-replication 또는 final replication and switch over 프로세스로 마이그레이션을 여러 번 실행할 수 있습니다.

기존 서버의 프로세스는 최종 마이그레이션 중에만 중지해야 합니다.

최종이 아닌 모든 복제 작업에 **--prepare** 파라미터를 추가하여 레거시 서버에서 서비스가 자동으로 중지되는 것을 방지하십시오. 예:

```
mgradm migrate podman <oldserver.fqdn> --prepare
```

절차: 레거시 서버에서 초기 준비

1. Uyuni 서비스를 중지합니다.

```
spacewalk-service stop
```

2. PostgreSQL 서비스를 중지합니다.

```
systemctl stop postgresql
```

4.1.2.2.2. SSH Connection Preparation

절차: SSH 연결 준비

1. 새 2026.01 서버에 **root**에 대한 SSH 키가 있는지 확인합니다. 키가 없는 경우 다음 명령을 사용하여 만듭니다.

```
ssh-keygen -t rsa
```

2. 새 서버 호스트에서 비밀번호를 묻지 않는 레거시 서버에 연결할 수 있도록 SSH 구성 및 에이전트가 준비되어 있어야 합니다.

```
eval $(ssh-agent); ssh-add
```



비밀번호를 묻지 않고 연결을 설정하기 위해 마이그레이션 스크립트는 새 서버에서 실행 중인 SSH 에이전트를 활용합니다. 에이전트가 아직 활성화되어 있지 않은 경우, **eval \$(ssh-agent)**를 실행하여 에이전트를 시작합니다. 그런 다음 **ssh-add**를 실행한 후 개인 키의 경로를 입력하여 실행 중인 에이전트에 SSH 키를 추가합니다. 이 프로세스 중에는 개인 키의 비밀번호를 입력하라는 메시지가 표시됩니다.

3. **ssh-copy-id**를 사용하여 레거시 Uyuni 서버(<oldserver.fqdn>)에 공개 SSH 키를 복사합니다. <oldserver.fqdn>을 레거시 서버의 FQDN으로 바꿉니다.

```
ssh-copy-id <oldserver.fqdn>
```

SSH 키가 레거시 서버의 **~/.ssh/authorized_keys** 파일에 복사됩니다. 자세한 내용은 **ssh-copy-id** 사용자 지정 페이지에서 확인할 수 있습니다.

4. 새 서버에서 레거시 Uyuni 서버로 SSH 연결을 설정하여 비밀번호가 필요하지 않은지 확인합니다. 호스트 지문에도 문제가 없어야 합니다. 문제가 있는 경우 **~/.ssh/known_hosts** 파일에서 기존 지문을 제거합니다. 그런 다음 다시 시도합니다. 지문은 로컬 **~/.ssh/known_hosts** 파일에 저장됩니다.

4.1.2.2.3. 마이그레이션 수행

레거시 Uyuni에서 컨테이너화된 Uyuni(으)로 마이그레이션을 계획할 때는 대상 인스턴스가 레거시 설정의 사양을 충족하거나 초과하는지 확인하십시오. 여기에는 메모리(RAM), CPU 코어, 스토리지 및 네트워크 대역폭이 포함되지만, 이에 국한되지 않습니다.

보안을 위해 강화된 Uyuni 서버 호스트는 **/tmp** 폴더의 파일 실행을 제한할 수 있습니다. 이러한 경우 해결 방법으로 **TMPDIR** 환경 변수를 다른 기준 경로로 내보낸 다음, **mgradm**을 실행합니다.

예:

```
export TMPDIR=/path/to/other/tmp
```

Uyuni 업데이트에서는 이 해결 방법이 불필요하도록 도구가 변경될 예정입니다.

사용자 정의 영구 스토리지 구성

영구 스토리지 구성은 선택 사항이지만, 컨테이너 디스크가 가득 찬 상황에 심각한 문제를 방지할 수 있는 유일한 방법입니다. **mgr-storage-server** 도구를 사용하여 사용자 지정 영구 스토리지를 구성하는 것이 적극 권장됩니다.

자세한 내용은 **mgr-storage-server --help**를 참조하십시오. 이 도구는 컨테이너 스토리지 및 데이터베이스 볼륨 생성을 간소화합니다.

다음 방법으로 명령 사용:

```
mgr-storage-server <storage-disk-device> [<database-disk-device>]
```



장치에는 파일 시스템이 없어야 합니다. 저장 장치에 파일 시스템이 있으면 명령이 중단됩니다.

예:

```
mgr-storage-server /dev/nvme1n1 /dev/nvme2n1
```

이 명령은 **/var/lib/containers/storage/volumes**에 영구 스토리지 볼륨을 생성합니다.

 자세한 내용은 다음을 참조하십시오.

- Installation-and-upgrade > Container-management
- Administration > Troubleshooting

マイグレーション 수행

1. 새 Uyuni 서버를 설치하려면 다음 명령을 실행합니다. <**oldserver.fqdn**>은 레거시 서버의 FQDN으로 바꿉니다.

```
mgradm migrate podman <oldserver.fqdn>
```

2. 신뢰할 수 있는 SSL CA 인증서를 마이그레이션합니다.

Migration of the Certificates

RPM의 일부로 설치되고 레거시 Uyuni의 **/usr/share/pki/trust/anchors** 디렉토리에 저장된 신뢰할 수 있는 SSL CA 인증서는 마이그레이션되지 않습니다. SUSE는 컨테이너에 RPM 패키지를 설치하지 않으므로 관리자는 마이그레이션 후 레거시 설치에서 이러한 인증서 파일을 수동으로 마이그레이션해야 합니다.

Procedure: Migrating the Certificates

1. 레거시 서버에서 새 서버로 파일을 복사합니다. 예를 들어 **/local/ca.file**일 수 있습니다.
2. 다음 명령을 사용하여 파일을 컨테이너에 복사합니다.

```
mgrctl cp /local/ca.file server:/etc/pki/trust/anchors/
```

マイグレーション 마무리

 **mgradm migrate** 명령을 성공적으로 실행한 후에도 모든 클라이언트의 Salt 설정은 계속해서 기존 레거시 서버를 가리킵니다.

 새 2026.01 서버로 리디렉션하려면, 기존 서버와 동일한 FQDN 및 IP 주소를 사용하도록 새 서버의 인프라 수준(DHCP 및 DNS)에서 이름을 변경해야 합니다.

 마이그레이션에 문제가 발생할 경우 기존 시스템을 재시작할 수 있습니다. 루트 권한으로 다음 명령을 사용하여 PostgreSQL 및 spacewalk 서비스를 다시 시작합니다.

```
service postgresql start  
spacewalk-service start
```

4.1.2.3. Kubernetes 준비

mgradm migrate 명령으로 마이그레이션을 실행하기 전에, 특히 마이그레이션 작업이 컨테이너를 처음부터 시작한다는 점을 고려하여 **영구 볼륨**을 필수로 미리 정의해야 합니다.

자세한 내용은 **Installation-and-upgrade > Container-management**의 설치 섹션에서 이러한 볼륨 준비 방법을 참조하십시오.

4.1.2.4. 마이그레이션

다음 명령을 실행하여 <**oldserver.fqdn**>을 레거시 서버의 적절한 FQDN으로 바꿔 새 Uyuni 서버를 설치합니다.

```
mgradm migrate podman <oldserver.fqdn>
```

또는

```
mgradm migrate kubernetes <oldserver.fqdn>
```



mgradm migrate 명령이 성공적으로 실행된 후에도 모든 클라이언트의 Salt 설정이 계속해서 레거시 서버를 가리킵니다. 이러한 설정을 새 서버로 리디렉션하려면 인프라 수준(DHCP 및 DNS)에서 새 서버에서 레거시 서버와 동일한 FQDN 및 IP 주소를 사용하도록 이름을 변경해야 합니다.

4.1.3. Uyuni 서버 업그레이드

업그레이드 명령을 실행하기 전에 호스트 운영 체제를 업데이트해야 합니다. 호스트 운영 체제를 업데이트하면 **mgradm** 도구와 같은 Uyuni 도구도 함께 업데이트됩니다.

절차: 서버 업그레이드

1. Refresh software repositories with **zypper**:

```
zypper ref
```

2. Apply available updates with **transactional-update**:

```
transactional-update
```

3. 업데이트가 완료되면 **reboot**를 수행합니다.

4. The Uyuni Server container can be updated using the following command:

```
mgradm upgrade podman
```

이 명령은 컨테이너의 상태를 최신 상태로 가져오고 서버를 다시 시작합니다.

5. 사용하지 않는 컨테이너 이미지를 정리하여 디스크 공간을 확보합니다.

```
podman image prune -a
```

타사 SSL 인증서로 업그레이드

타사 인증서를 사용하는 경우, 데이터베이스 컨테이너에는 다음의 SAN(Subject Alternate Names)이 포함된 SSL 인증서가 필요합니다.

- db
- reportdb
- 외부에 노출되는 정규화된 도메인 이름

기본 컨테이너와 데이터베이스 컨테이너 모두에 동일한 인증서를 사용할 수 있지만 해당 SAN도 있어야 합니다.

In order to pass the new certificate to the upgrade command, use the **--ssl-db-ca-root**, **--ssl-db-cert** and **--ssl-db-key** parameters.

특정 버전으로 업그레이드

If you do not specify the tag parameter, it will default to upgrading to the most recent version. To upgrade to a specific version, provide the tag parameter with the desired image tag.

업그레이드 명령과 해당 파라미터에 대한 자세한 내용은 다음 명령을 참조하십시오.

```
mgradm upgrade podman -h
```

air-gapped 설치의 경우, 먼저 컨테이너 RPM 패키지를 업그레이드한 후 **mgradm** 명령을 실행합니다.

4.1.3.1. 데이터베이스 백업 볼륨

mgradm migration 또는 **mgradm upgrade**를 사용한 서버 마이그레이션 또는 업그레이드는 데이터베이스 백업과 함께 볼륨을 생성할 수 있습니다.

PostgreSQL 데이터베이스 버전을 업그레이드할 때는 기존 데이터베이스를 별도의 위치에 백업해 두어야 합니다. 이를 위해 **mgradm**이 **var-pgsql-backup** 볼륨을 동적으로 생성합니다. 마이그레이션 또는 업그레이드가 완료되고 새 시스템의 정상 작동을 사용자가 확인한 후에는 이 볼륨을 안전하게 제거할 수 있습니다.

4.2. 프록시

4.2.1. Migrating the Uyuni Proxy to openSUSE Tumbleweed

This page describes how to migrate a Uyuni Proxy host from openSUSE Leap Micro 5.5 to a fresh openSUSE Tumbleweed installation using the proxy administration tool **mgrpctx**.

 This guide was tested on Tumbleweed only. There is no known reason it wouldn't

- work on other supported bases, but always validate in a test environment before production.

4.2.1.1. Overview of the Proxy Migration Process

You will:

- Save proxy configuration from the old system (including Apache/Squid tuning).
- Reinstall the host with openSUSE Tumbleweed.
- Re-register the host using the system reactivation key.
- Install **mgrpxy** (and Podman if needed).
- Restore configuration and run **mgrpxy install podman** with optional tuning files.

4.2.1.2. Requirements and Considerations

- Keep the same hostname/FQDN and IP when possible so the server and clients interact with the proxy as before.
- Ensure you have the “system reactivation key” for the existing proxy system (UI: Systems > select the proxy > Details > Reactivation).
- Ensure SSH/scp access to move configuration archives off and onto the machine.

4.2.1.3. Migration Procedure

4.2.1.3.1. Step 1: Save Proxy Configuration and Tuning Files

Procedure: Save Proxy Configuration and Tuning Files

1. Copy the Uyuni proxy configuration directory to a safe location:

```
scp -r /etc/uyuni <USER>@<HOST>:/some/where/safe/
```

2. Identify Apache and Squid tuning files currently in use by the legacy proxy services:

```
systemctl cat uyuni-proxy-httpd.service | grep EXTRA_CONF= | sed 's/.*=-\v\[^\:\]\+\:\.*\//'
systemctl cat uyuni-proxy-squid.service | grep EXTRA_CONF= | sed 's/.*=-\v\[^\:\]\+\:\.*\//'
```

3. Copy those tuning files to the same safe location as well.



Typical default paths after you copy them back will be:

- Apache tuning: `/etc/uyuni/proxy/apache.conf`

- Squid tuning: /etc/uyuni/proxy/squid.conf

4.2.1.3.2. Step 2: Reinstall the Host with openSUSE Tumbleweed

Procedure: Reinstalling the Host with openSUSE Tumbleweed

1. Reinstall the machine with openSUSE Tumbleweed (server profile recommended).
2. Set the same hostname/FQDN and IP as before when possible.

4.2.1.3.3. Step 3: Re-register the Host with the Reactivation Key

Procedure: Re-registering the Host with the Reactivation Key

1. From the Uyuni Web UI, obtain the system reactivation key for the existing proxy system record (Systems > Details > Reactivation).
2. Bootstrap/re-register the Tumbleweed host using that reactivation key so it claims the existing system entry.
 - i** Use your standard bootstrapping process for Tumbleweed hosts in your environment (for example, the bootstrap script or your configuration management), ensuring the reactivation key is applied.

4.2.1.3.4. Step 4: Install Uyuni Proxy Tools and Podman

Procedure: Installing Proxy Tools and Podman

1. Add the Uyuni Stable repository and install tools:

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/images/repo/Uyuni-Proxy-POOL-x86_64-Media1 uyuni-proxy-stable
zypper ref
zypper in mgrpxy mgrctl mgrpxy-bash-completion mgrctl-bash-completion
```

2. Ensure Podman is installed (required to run containers):

```
zypper in podman
```

4.2.1.3.5. Step 5: Restore Configuration and Install the Proxy

Procedure: Restoring Configuration and Install the Proxy

1. Copy back the saved configuration directory to the new host:

```
scp -r <USER>@<HOST>:/some/where/safe/uyuni /etc/
```

- If you saved Apache/Squid tuning files, place them at the expected default paths or note their locations for parameters in the next command:

```
# Default paths expected by mgrpxy parameters (adjust/move your files accordingly)
# Apache tuning: /etc/uyuni/proxy/apache.conf
# Squid tuning: /etc/uyuni/proxy/squid.conf
```

- Run the proxy installation with Podman. If you do not use tuning files, omit the corresponding parameters:

```
# With tuning files
mgrpxy install podman \
    --tuning-httpd /etc/uyuni/proxy/apache.conf \
    --tuning-squid /etc/uyuni/proxy/squid.conf

# If you have no tuning files, remove the tuning parameters:
# mgrpxy install podman
```



In an upcoming release, if tuning files are placed at the default paths noted above, the explicit parameters will not be required.

4.2.1.3.6. Step 6: Verify the Proxy

Procedure: Verifying the Proxy

- Check containers are running:

```
mgrctl ps
# or
podman ps
```

- Confirm the proxy appears healthy in the Uyuni Web UI and that clients using this proxy operate normally.

4.2.1.4. 문제 해결

- If Podman was missing, install it and rerun the **mgrpxy install** step.
- Verify the host's time, hostname, and IP match expectations.
- If the host did not reattach to the existing system record, confirm you used the correct reactivation key and repeat the bootstrap.

4.2.2. 레거시 프록시를 컨테이너로 마이그레이션

컨테이너화된 프록시는 이제 일련의 systemd 서비스로 관리됩니다. 컨테이너화된 프록시를 관리하려면 **mgrpxy** 도구를 사용하십시오.

이 섹션에서는 [command] **mgrpxy** 도구를 사용하여 레거시 **systemd** 프록시로부터 마이그레이션하는 데 도움을 줍니다.

Uyuni의 이전 버전에서 2026.01(으)로의 인플레이스 마이그레이션은 호스트 OS가 openSUSE Leap에서 openSUSE Leap Micro로 변경됨에 따라 계속해서 지원되지 않습니다.



기존 연락 프로토콜은 Uyuni 2026.01 이상에서 더 이상 지원되지 않습니다. 기존 프록시를 포함한 모든 기존 클라이언트를 Salt로 마이그레이션한 후 이전 Uyuni 릴리스에서 2026.01(으)로 마이그레이션해야 합니다.

4.2.2.1. Migrate From Legacy to Containerized Proxy With Systemd

4.2.2.1.1. 프록시 구성 생성

절차: 프록시 구성 생성

1. Uyuni 서버 Web UI에 로그인합니다.
2. 왼쪽 탐색에서 **시스템 > 프록시 구성**을 선택합니다.
3. 프록시 FQDN을 입력합니다. 원래 프록시 호스트와 동일한 FQDN을 사용합니다.
4. 서버 FQDN을 입력합니다.
5. Enter the Proxy port number. We recommend using the default port of 8022.
6. 인증서와 개인 키는 서버 컨테이너 호스트의 `/var/lib/containers/storage/volumes/root/_data/ssl-build/`에 있습니다.
 - RHN-ORG-TRUSTED-SSL-CERT
 - RHN-ORG-PRIVATE-SSL-KEY
7. 다음을 사용하여 인증서와 키를 컴퓨터로 복사합니다.

```
scp root@uyuni-server-example.com:/root/ssl-build/RHN-ORG-PRIVATE-SSL-KEY .
scp root@uyuni-server-example.com:/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT .
```

8. [파일 선택]을 선택하고 로컬 컴퓨터에서 인증서를 찾습니다.
9. [파일 선택]을 선택하고 로컬 컴퓨터에서 개인 키를 찾습니다.
10. CA 비밀번호를 입력합니다.
11. [생성]을 클릭합니다.

4.2.2.1.2. 프록시 구성 새 호스트로 전송

절차: 프록시 구성 이전

1. 서버에서 프록시 구성이 포함된 생성된 tar.gz 파일을 새 프록시 호스트로 전송합니다.

```
scp config.tar.gz <uyuni-proxy-FQDN>:/root/
```

2. 다음 단계를 실행하기 전에 레거시 프록시를 비활성화합니다.

```
spacewalk-proxy stop
```

3. 다음을 사용하여 새 프록시를 배포합니다.

```
systemctl start uyuni-proxy-pod
```

4. 다음을 사용하여 새 프록시를 활성화합니다.

```
systemctl enable --now uyuni-proxy-pod
```

5. `podman ps`를 실행하여 모든 컨테이너가 있고 실행 중인지 확인합니다.

```
proxy-salt-broker
proxy-htpd
proxy-tftpd
proxy-squid
proxy-ssh
```

4.2.2.2. Migrate Uyuni Proxy to Uyuni 2026.01 Containerized Proxy

절차: Uyuni 컨테이너화된 프록시를 Uyuni 2026.01 새 컨테이너화된 프록시로 マイグ레이션

1. 새 시스템을 부팅하고 openSUSE Leap Micro 6.1 설치를 시작합니다.
2. 설치를 완료합니다.
3. 시스템 업데이트:

```
transactional-update --continue
```

4. **mgrpxy**와 선택적으로 **mgrpxy-bash-completion**을 설치합니다.

```
transactional-update pkg install mgrpxy mgrpxy-bash-completion
```

5. 재부팅합니다.
6. **.tar.gz** 프록시 구성 파일을 호스트에 복사합니다.

4.2.2.3. Web UI를 사용하여 패키지 설치

mgrpxy 및 **mgrpxy-bash-completion** 패키지는 미니언이 부트스트랩되어 서버에 등록된 후 웹 UI를 통해 설치할 수도 있습니다.

절차: Web UI를 사용하여 패키지 설치

1. 설치가 완료되면 관리자 > 설정 마법사 → 제품 페이지에서 SLE Micro 6.1 상위 채널과 프록시 하위 채널을 추가하고 동기화합니다.
2. Web UI에서 시스템 > 활성화 키로 이동하여 동기화된 SLE Micro 6.1 채널에 연결된 활성화 키를 만듭니다.
3. 시스템 > 부트스트랩 페이지를 사용하여 시스템을 미니언으로 부트스트랩합니다.
4. 새 머신이 온보딩되어 시스템 목록에 표시되면 시스템을 선택하고 시스템 세부 정보 > 패키지 설치 페이지로 이동합니다.
5. grpctl 및 grpctl-bash-completion 패키지를 설치합니다.
6. 시스템을 재부팅합니다.

4.2.2.4. spacecmd 및 자체 서명 인증서를 사용하여 프록시 구성 생성

spacecmd를 사용하여 프록시 구성 생성할 수 있습니다.

절차: spacecmd 및 자체 서명 인증서를 사용하여 프록시 구성 생성

1. 컨테이너 호스트에 SSH로 연결합니다.
2. 서버 및 프록시 FQDN을 바꾸는 다음 명령을 실행합니다.

```
mgrctl exec -ti 'spacecmd proxy_container_config_generate_cert -- dev-pxy.example.com
dev-srv.example.com 2048 email@example.com -o /tmp/config.tar.gz'
```

3. 생성된 구성을 프록시에 복사합니다.

```
mgrctl cp server:/tmp/config.tar.gz .
```

4. 다음을 사용하여 프록시를 배포합니다.

```
mgrpxy install podman config.tar.gz
```

4.2.2.5. spacecmd 및 사용자 지정 인증서를 사용하여 프록시 구성 생성

spacecmd를 사용하여 기본 자체 서명 인증서가 아닌 사용자 정의 인증서에 대해 프록시 구성 생성할 수 있습니다.

 2GB는 기본 프록시 squid 캐시 크기입니다. 사용자의 환경에 적합하도록 조정해야 합니다.

절차: spacecmd 및 사용자 정의 인증서를 사용하여 프록시 구성 생성

1. 서버 컨테이너 호스트에 SSH로 연결합니다.
2. 서버 및 프록시 FQDN을 바꾸는 다음 명령을 실행합니다.

```
for f in ca.crt proxy.crt proxy.key; do
    mgrctl cp $f server:/tmp/$f
done
mgrctl exec -ti 'spacecmd proxy_container_config -- -p 8022 pxy.example.com'
```

```
srv.example.com 2048 email@example.com /tmp/ca.crt /tmp/proxy.crt /tmp/proxy.key -o
/tmp/config.tar.gz'
```

3. 생성된 구성을 프록시에 복사합니다.

```
mgrctl cp server:/tmp/config.tar.gz .
```

4. 다음을 사용하여 프록시를 배포합니다.

```
mgrpxy install podman config.tar.gz
```

4.2.3. Uyuni 프록시 업그레이드

업그레이드 명령을 실행하기 전에 호스트 운영 체제를 업데이트해야 합니다. 호스트 운영 체제를 업데이트하면 **mgrpxy** 도구와 같은 Uyuni 도구도 함께 업데이트됩니다.

절차: 프록시 업그레이드

1. Refresh software repositories with **zypper**:

```
zypper ref
```

2. Apply available updates with **transactional-update**:

```
transactional-update
```

3. 업데이트가 완료되면 **reboot**를 수행합니다.

4. 다음 명령을 사용하여 **podman**에서 실행 중인 2026.01 프록시 컨테이너를 업데이트할 수 있습니다.

```
mgrpxy upgrade podman
```

Or, those running on a Kubernetes cluster can update using:

```
mgrpxy upgrade kubernetes
```

5. **podman**에서, 사용하지 않는 컨테이너 이미지를 정리하여 디스크 공간을 확보합니다.

```
podman image prune -a
```

Kubernetes에서는 이미지 정리가 자동으로 처리되거나 Kubernetes 배포판에 따라 다릅니다.



If you do not specify the tag parameter when upgrading to specific version, it will default to upgrading to the most recent version. To upgrade to a specific version, provide the tag parameter with the desired image tag.



일반적인 상황에서는 일관성을 유지하기 위해 모든 프록시 컨테이너에 동일한 태그를 사용하는 것이 좋습니다.

For air-gapped installations, first upgrade the container RPM packages, then run the **mgrpxy upgrade podman** command.

4.3. 클라이언트

4.3.1. 클라이언트 업그레이드

클라이언트는 기본 운영 체제의 버전 관리 시스템을 사용합니다. SUSE 운영 체제를 사용하는 클라이언트의 경우, Uyuni Web UI 내에서 업그레이드할 수 있습니다.

클라이언트 업그레이드에 대한 자세한 정보는 [Client-configuration > Client-upgrades](#)에서 참조하십시오.

장 5. Basic Server and Proxy Management

5.1. Custom YAML Configuration and Deployment with mgradm

배포 중에 **mgradm** 도구가 사용할 수 있는 사용자 지정 **mgradm.yaml** 파일을 만들 수 있는 옵션이 제공됩니다.

 명령줄 파라미터 또는 **mgradm.yaml** 구성 파일을 사용하여 기본 변수를 제공하지 않은 경우 **mgradm**은 기본 변수를 묻는 메시지를 표시합니다.

For security, **using command line parameters to specify passwords should be avoided**. Use a configuration file with proper permissions instead.

Procedure: Deploying the Uyuni Container with Podman Using a Custom Configuration File

1. 다음 예제와 유사하게 **mgradm.yaml**이라는 이름의 구성 파일을 준비합니다.

```
# 데이터베이스 비밀번호. 기본적으로 무작위로 생성됨
db:
    password: MySuperSecretDBPass

# CA 인증서의 비밀번호
ssl:
    password: MySuperSecretSSLPASSWORD

# SUSE 고객 센터 자격 증명
scc:
    user: ccUsername
    password: ccPassword

# 조직 이름
organization: YourOrganization

# 알림을 전송하는 이메일 주소
emailFrom: notifications@example.com

# 관리자 계정 세부 정보
admin:
    password: MySuperSecretAdminPass
    login: LoginName
    firstName: Admin
    lastName: Admin
    email: email@example.com
```

2. 터미널에서 루트 권한으로 다음 명령을 실행합니다. 서버의 FQDN 입력은 선택 사항입니다.

```
mgradm -c mgradm.yaml install podman <FQDN>
```

 컨테이너를 sudo 또는 루트로 배포해야 합니다. 이 단계를 건너뛰면 터미널에 다음 오류가 표시됩니다.

```
INF Setting up uyuni network
```

```
9:58AM INF Enabling system service
9:58AM FTL Failed to open /etc/systemd/system/uyuni-server.service
for writing
error="open /etc/systemd/system/uyuni-server.service: permission
denied"
```

3. 배포가 완료될 때까지 기다립니다.
4. 브라우저를 열고 서버의 FQDN 또는 IP 주소로 이동합니다.

5.2. 컨테이너 시작 및 중지

다음 명령을 사용하여 Uyuni 2026.01 서버 컨테이너를 재시작, 시작 및 중지할 수 있습니다.

Uyuni 2026.01 서버를 **restart**하려면 다음 명령을 실행합니다.

```
# mgradm restart
5:23PM INF Welcome to mgradm
5:23PM INF Executing command: restart
```

서버를 **start**하려면 다음 명령을 실행합니다.

```
# mgradm start
5:21PM INF Welcome to mgradm
5:21PM INF Executing command: start
```

서버를 **stop**하려면 다음 명령을 실행합니다.

```
# mgradm stop
5:21PM INF Welcome to mgradm
5:21PM INF Executing command: stop
```

5.3. Containers used by Uyuni

Below is a list of containers used by Uyuni 2026.01.

표 10. Server Containers

| 컨테이너 이름 | 설명 |
|--------------------------|--------------------------|
| uyuni-server | 기본 제품 컨테이너 |
| uyuni-db | 제품의 데이터베이스 컨테이너 |
| uyuni-hub-xmlrpc | 허브 배포의 XML-RPC 게이트웨이 |
| uyuni-server-attestation | 서버 COCO 증명 |
| uyuni-saline | Salt 옵저버빌리티의 Saline 컨테이너 |
| uyuni-server-migration | マイグ레이션 ヘルパー コンテイナー |

표 11. Proxy Containers

| Container Name | Description |
|-------------------------|--|
| uyuni-proxy-htpd | Main proxy container handling all HTTP communication |
| uyuni-proxy-squid | Squid cache |
| uyuni-proxy-salt-broker | Salt forwarder |
| uyuni-proxy-ssh | SSH forwarder |
| uyuni-proxy-tftpd | TFTPd to HTTP translator and forwarder |

5.4. 영구 컨테이너 볼륨

컨테이너 내에서 수행한 수정 사항은 유지되지 않습니다. 영구 볼륨 외부에서 변경한 내용은 모두 삭제됩니다. 아래는 Uyuni 2026.01에 대한 영구 볼륨 목록입니다.

기본 볼륨 위치를 사용자 정의하려면 Podman을 처음 시작하기 전에 **podman volume create** 명령을 사용하여 필요한 볼륨을 생성해야 합니다.



이 테이블이 Helm 차트와 systemctl 서비스 정의에 설명된 볼륨 매핑과 세부적으로 일치하는지 확인합니다.

5.4.1. 서버

The following volumes are stored under the **Podman** default storage location on the server.

표 12. 영구 볼륨: Podman 기본 스토리지

| 볼륨 이름 | 볼륨 디렉토리 |
|-------------|--------------------------------------|
| Podman 스토리지 | /var/lib/containers/storage/volumes/ |

표 13. 영구 볼륨: 루트

| 볼륨 이름 | 볼륨 디렉토리 |
|-------|---------|
| root | /root |

표 14. 영구 볼륨: var/

| 볼륨 이름 | 볼륨 디렉토리 |
|-------------|---------------------|
| var-cobbler | /var/lib/cobbler |
| var-salt | /var/lib/salt |
| var-pgsql | /var/lib/pgsql/data |

| 볼륨 이름 | 볼륨 디렉토리 |
|------------------|-----------------------|
| var-pgsql-backup | /var/lib/pgsql-backup |
| var-cache | /var/cache |
| var-spacewalk | /var/spacewalk |
| var-log | /var/log |

표 15. 영구 볼륨: srv/

| 볼륨 이름 | 볼륨 디렉토리 |
|---------------------|-----------------------|
| srv-salt | /srv/salt |
| srv-www | /srv/www/ |
| srv-tftpboot | /srv/tftpboot |
| srv-formulametadata | /srv/formula_metadata |
| srv-pillar | /srv/pillar |
| srv-susemanager | /srv/susemanager |
| srv-spacewalk | /srv/spacewalk |

표 16. 영구 볼륨: etc/

| Volume Name | Volume Directory |
|---------------------|---|
| etc-apache2 | /etc/apache2 |
| etc-rhn | /etc/rhn |
| etc-systemd-multi | /etc/systemd/system/multi-user.target.wants |
| etc-systemd-sockets | /etc/systemd/system/sockets.target.wants |
| etc-salt | /etc/salt |
| etc-sssd | /etc/sssd |
| etc-tomcat | /etc/tomcat |
| etc-cobbler | /etc/cobbler |
| etc-sysconfig | /etc/sysconfig |
| etc-postfix | /etc/postfix |
| ca-cert | /etc/pki/trust/anchors |

표 17. 영구 볼륨: run/

| 볼륨 이름 | 볼륨 디렉토리 |
|-----------------|------------------|
| run-salt-master | /run/salt/master |

5.4.2. 프록시

The following volumes are stored under the **Podman** default storage location on the proxy.

표 18. 영구 볼륨: Podman 기본 스토리지

| 볼륨 이름 | 볼륨 디렉토리 |
|-------------|--------------------------------------|
| Podman 스토리지 | /var/lib/containers/storage/volumes/ |

표 19. 영구 볼륨: srv/

| Volume Name | Volume Directory |
|----------------------|------------------|
| uyuni-proxy-tftpboot | /srv/tftpboot |

표 20. 영구 볼륨: var/

| Volume Name | Volume Directory |
|-------------------------|------------------|
| uyuni-proxy-rhn-cache | /var/cache/rhn |
| uyuni-proxy-squid-cache | /var/cache/squid |

5.5. mgr-storage-server 및 mgr-storage-proxy 이해하기

mgr-storage-server 및 **mgr-storage-proxy**는 Uyuni과(와) 함께 제공되는 보조 스크립트입니다.

이는 Uyuni 서버 및 프록시를 위한 스토리지를 구성하도록 설계되었습니다.

이 스크립트는 디스크 장치를 인수로 받습니다. **mgr-storage-proxy**에는 스토리지 디스크 장치를 위한 단일 인수가 필요합니다. **mgr-storage-server**에는 스토리지 디스크 장치가 필요하며, 선택적으로 전용 데이터베이스 디스크 장치를 위한 두 번째 인수를 받아들일 수 있습니다. 일반 스토리지와 데이터베이스 스토리지가 동일한 디스크에 존재할 수 있지만, 더 나은 성능과 쉬운 관리를 보장하기 위해 데이터베이스를 전용 고성능 디스크에 배치하는 것이 좋습니다.

5.5.1. 이러한 도구의 기능

mgr-storage-server 및 **mgr-storage-proxy**는 모두 표준 스토리지 설정 작업을 수행합니다.

- 제공된 저장 장치를 확인합니다.
- 장치가 비어 있고 사용에 적합한지 확인합니다.
- 지정된 장치에 XFS 파일 시스템을 생성합니다.
- 데이터 마이그레이션을 위해 장치를 임시로 마운트합니다.

- 관련 저장소 디렉토리를 새 장치로 이동합니다.
- /etc/fstab에 항목을 생성하여 부팅 시 저장소가 자동으로 마운트되도록 합니다.
- 장치를 최종 위치에 다시 마운트합니다.

표 21. 도구별 추가 동작

| | |
|---------------------------|---|
| mgr-storage-server | <ul style="list-style-type: none"> 선택적으로 데이터베이스 저장을 위한 별도의 장치를 지원합니다. マイグ레이션 중 SUSE Manager 서비스를 중지하고, 이후에 다시 시작합니다. podman 볼륨 디렉토리 <code>/var/lib/containers/storage/volumes</code>을 준비된 저장소로 이동하고, 선택적으로 <code>/var/lib/containers/storage/volumes/var-pgsql</code>을 준비된 데이터베이스 저장소로 이동합니다. |
| mgr-storage-proxy | <ul style="list-style-type: none"> 프록시 저장소에만 집중합니다(데이터베이스 저장소 지원 안 함). マイグ레이션 중 프록시 서비스를 중지하고 다시 시작합니다. podman 볼륨 디렉토리 <code>/var/lib/containers/storage/volumes</code>을 준비된 저장소로 이동합니다. |



- 두 도구 모두 표준 Linux 스토리지 작업을 자동화합니다. Linux 관리자가 수동으로 수행하는 작업 이상의 숨겨진 또는 사용자 정의 논리는 존재하지 않습니다.

5.5.2. 이러한 도구가 제공하지 않는 기능

- LVM 볼륨을 생성하거나 관리하지 않습니다.
- RAID 또는 복잡한 저장소 토플로지를 구성하지 않습니다.
- 설정 후 일반적인 Linux 도구를 사용하여 저장소를 관리하는 것을 방해하지 않습니다.
- 동적 크기 조정 또는 확장 기능은 제공하지 않습니다. 이러한 작업은 표준 Linux 저장소 도구를 사용하여 처리해야 합니다.

5.5.3. 설치 후 저장소 관리

저장소 구성이 완료되면 표준 Linux 명령을 사용하여 안전하게 관리할 수 있습니다.

5.5.3.1. Examples

목록 5. 예시 1: LVM 사용 시 저장소 확장

```
lvextend -L +10G /dev/your_vg/your_lv
xfs_growfs /var/lib/containers/storage/volumes
```

예시 2: 더 큰 디스크로 마이그레이션

- 새 디스크를 추가하고 포맷합니다.
- 일시적으로 마운트합니다.
- rsync를 사용하여 데이터를 복사합니다.

4. /etc/fstab를 업데이트합니다.

5. 올바른 위치에 다시 마운트합니다.

5.5.4. 사용해야 하는 경우 또는 사용하지 않아야 하는 경우



저장소 설정을 변경하기 전에 항상 백업을 수행하십시오.

- 이러한 도구는 초기 저장소 설정 시 또는 데이터 마이그레이션 및 /etc/fstab 업데이트를 처리할 것으로 예상되는 새 스토리지로 마이그레이션하는 **경우에만** 사용하십시오.
- 저장소 크기 조정 또는 확장을 위해 이 스크립트를 다시 실행하지 **마십시오**. 이러한 작업에는 표준 Linux 도구(예: lvextend, xfs_growfs)를 사용하십시오.

5.5.5. Summary

mgr-storage-server 및 **mgr-storage-proxy**는 표준 Linux 저장소 관행을 사용하여 Uyuni 구성 요소에 대한 초기 영구 저장소 설정을 자동화하는 데 도움을 줍니다. 이후 표준 저장소 관리에 제한을 두거나 간섭하지 않습니다.

설정 이후에는 익숙한 Linux 도구를 사용하여 저장소를 계속 관리합니다.



데이터베이스 볼륨이 가득 차면 시스템 운영에 심각한 문제가 발생할 수 있습니다. 디스크 사용량 알림 기능이 아직 컨테이너화된 환경에 적용되지 않았으므로, 사용자는 Grafana, Prometheus 등의 도구나 선호하는 다른 방법을 통해 podman 볼륨이 차지하는 디스크 공간을 직접 모니터링할 것을 권장합니다. 특히 **/var/lib/containers/storage/volumes/** 아래에 위치한 var-pgsql 볼륨에 각별히 유의해야 합니다.

장 6. GNU Free Documentation License

Copyright © 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

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. Secondly, 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".