Uyuni 2025.10

설치 및 업그레이드 가이드

UYUNI

장 1. Preface

Installation, Deployment and Upgrade + Uyuni 2025.10

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.

Publication Date: 2025-10-31

+ + + + + + + + + + + + + + + + + + +

차례

1. Preface 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 5
2.2.5. air-gapped 배포 6
2.2.6. 필수 네트워크 포트 6
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. Install Uyuni Server 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 22
3.2.3. Proxy conversion from client 27
3.2.4. Uyuni Proxy Deployment on K3s 30
4. 업그레이드 및 마이그레이션 32
4.1. 서버 32
4.1.1. Migrating the Uyuni Server to openSUSE Tumbleweed 32
4.1.2. Legacy Uyuni Server Migration to Container 35
4.1.3. Uyuni Server Upgrade 39
4.2. 프록시 40
4.2.1. Migrating the Uyuni Proxy to openSUSE Tumbleweed 40
4.2.2. Legacy Proxy Migration to Container 44
4.2.3. Uyuni Proxy Upgrade 47
4.3. 클라이언트 48
4.3.1. Upgrade Clients 48
5. Basic Server and Proxy Management 49
5.1. **mgradm**을 사용하여 사용자 지정 YAML 구성 및 배포 49
5.2. 컨테이너 시작 및 중지 50
5.3. Containers used by Uyuni 50
5.4. Persistent Container Volumes 51
5.4.1. 서버 51
5.4.2. 프록시 53
5.5. Understanding **mgr-storage-server** and **mgr-storage-proxy** 53
5.5.1. What these tools do 53
5.5.2. What these tools do **not** do 54
5.5.3. Post-installation storage management 54
5.5.4. When to use, or not use 55
5.5.5. Summary 55
6. GNU Free Documentation License 56

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-host에서 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.html에서 확인할 수 있습니다.

2.2.3. Reenable router advertisements

When the Uyuni is installed using **mgradm install podman** or **mgrpxy 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:

Procedure: Reenabling router advertisements

1. Create a file in **/etc/sysctl.d**, for example **99-ipv6-ras.conf**.
2. Add the following parameter and value to the file:

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

3. 재부팅합니다.

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

1. 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"
```

2. 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"
```

3. 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:

    a. 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

If you are on an internal network and do not have access to SUSE Customer Center, you can use an **Installation-and-upgrade › Container-deployment**.

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.

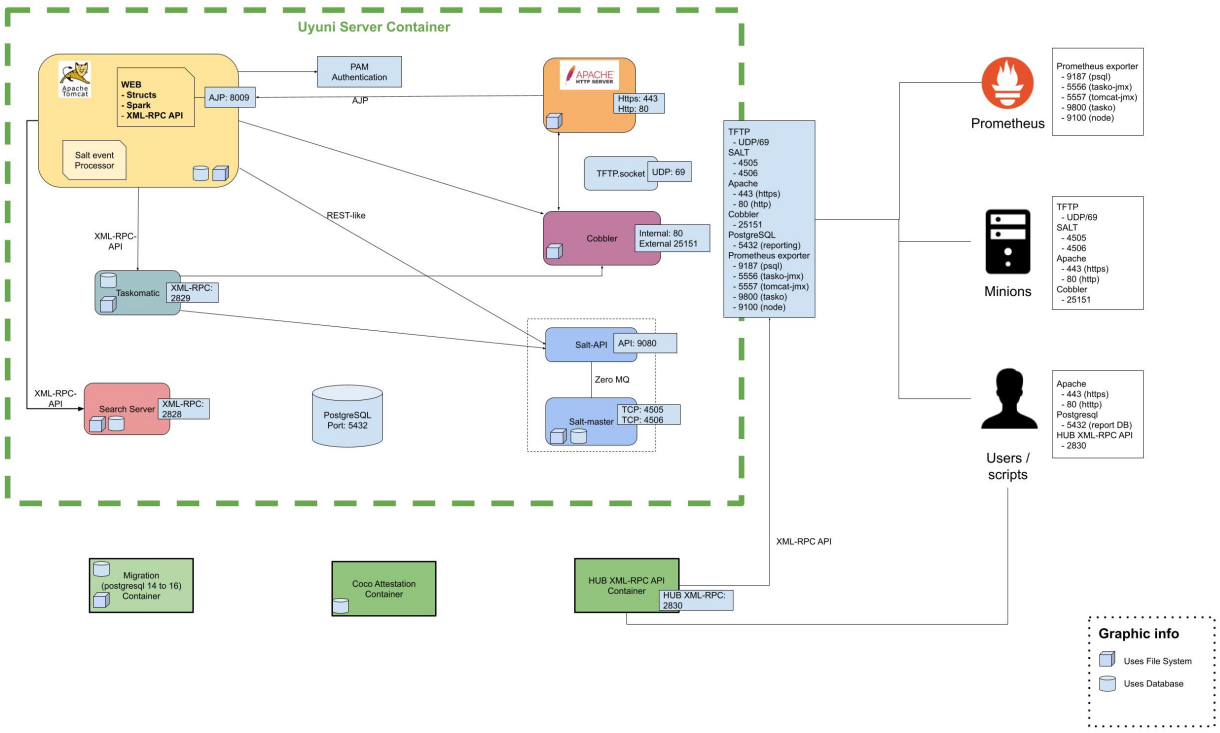
2.2.6. 필수 네트워크 포트

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

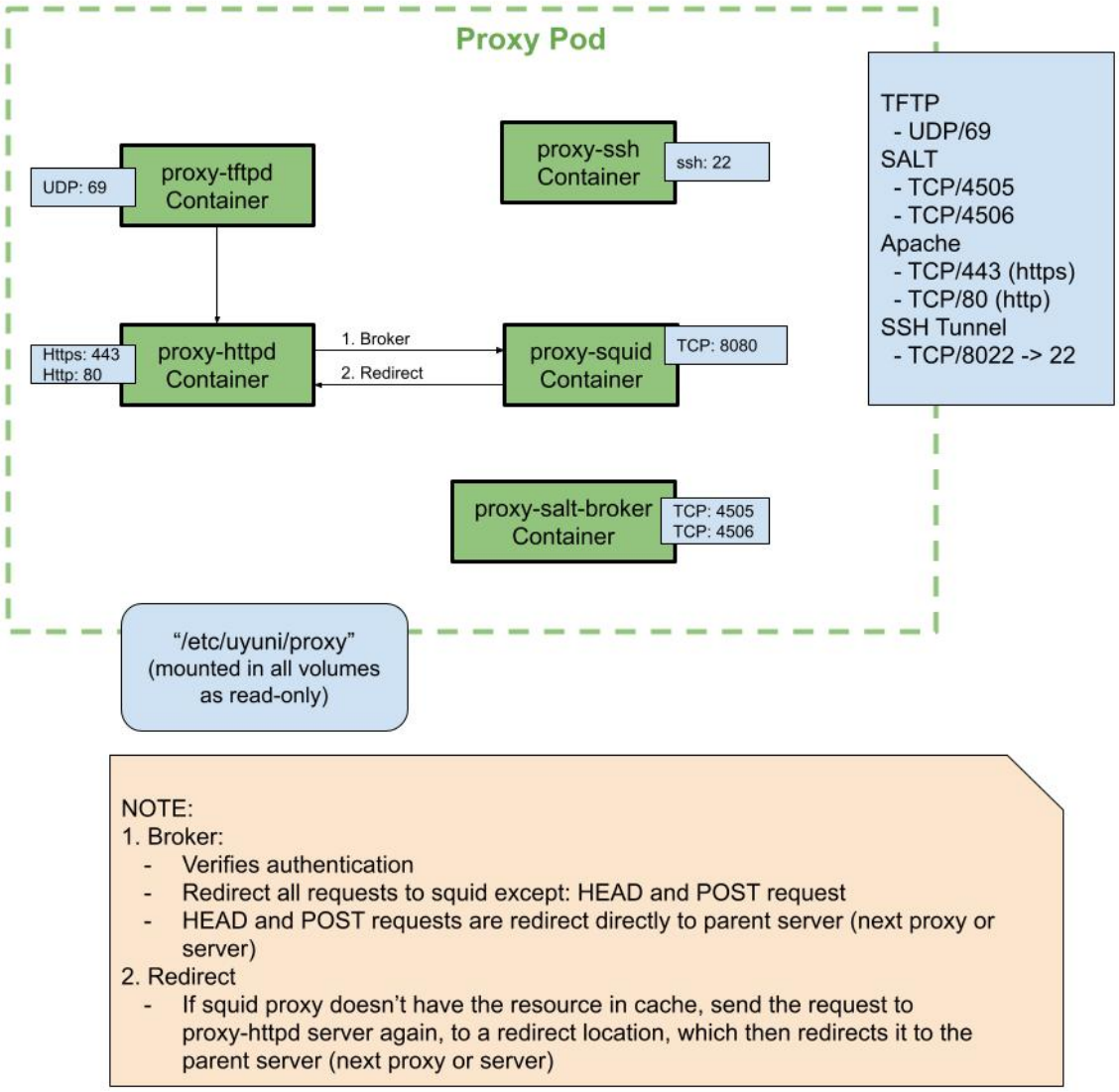
이러한 모든 포트를 열 필요는 없습니다. 사용 중인 서비스에 필요한 포트만 열면 됩니다.

2.2.6.1. Overview

2.2.6.1.1. 서버


Uyuni Server Container
WEB
- Structs
- Spark
- XML-RPC API
AJP: 8009
PAM Authentication
AJP
Salt event Processor
APACHE HTTP SERVER
Https: 443
Http: 80
TFTP.socket
UDP: 69
REST-like
XML-RPC-API
Cobbler
Internal: 80
External 25151
Taskomatic
XML-RPC: 2829
Salt-API
API: 9080
Zero MQ
Salt-master
TCP: 4505
TCP: 4506
XML-RPC-API
Search Server
XML-RPC: 2828
PostgreSQL Port: 5432
TFTP
- UDP/69
SALT
- 4505
- 4506
Apache
- 443 (https)
- 80 (http)
Cobbler
- 25151
PostgreSQL
- 5432 (reporting)
Prometheus exporter
- 9187 (psql)
- 5556 (tasko-jmx)
- 5557 (tomcat-jmx)
- 9800 (tasko)
- 9100 (node)
Prometheus
Minions
Users / scripts
Apache
- 443 (https)
- 80 (http)
Postgresql
- 5432 (report DB)
HUB XML-RPC API
- 2830
XML-RPC API
Migration (postgresql 14 to 16) Container
Coco Attestation Container
HUB XML-RPC API Container
HUB XML-RPC: 2830
Graphic info
Uses File System
Uses Database

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

Port number	Protocol	Used By	Notes
67	TCP/UDP	DHCP	Required only if clients are requesting IP addresses from the server.
69	TCP/UDP	TFTP	Required if server is used as a PXE server for automated client installation.

Port number	Protocol	Used By	Notes
80	TCP	HTTP	Required temporarily for some bootstrap repositories and automated installations.
443	TCP	HTTPS	Serves the Web UI, client, and server and proxy (**tftpsync**) requests.
4505	TCP	salt	Required to accept communication requests from clients. The client initiates the connection, and it stays open to receive commands from the Salt master.
4506	TCP	salt	Required to accept communication requests from clients. The client initiates the connection, and it stays open to report results back to the Salt master.
5432	TCP	PostgreSQL	Required to access the reporting database.
5556	TCP	Prometheus	Required for scraping Taskomatic JMX metrics.
5557	TCP	Prometheus	Required for scraping Tomcat JMX metrics.
9100	TCP	Prometheus	Required for scraping Node exporter metrics.
9187	TCP	Prometheus	Required for scraping PostgreSQL metrics.
9800	TCP	Prometheus	Required for scraping Taskomatic metrics.
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

Port number	Protocol	Used By	Notes
22			Only required if the user wants to manage the proxy host with Salt SSH.
67	TCP/UDP	DHCP	Required only if clients are requesting IP addresses from the server.
69	TCP/UDP	TFTP	Required if the server is used as a PXE server for automated client installation.
443	TCP	HTTPS	Web UI, client, and server and proxy (**tftpsync**) requests.

Port number	Protocol	Used By	Notes
4505	TCP	salt	Required to accept communication requests from clients. The client initiates the connection, and it stays open to receive commands from the Salt master.
4506	TCP	salt	Required to accept communication requests from clients. The client initiates the connection, and it stays open to report results back to the Salt master.
8022			Required for ssh-push and ssh-push-tunnel contact methods. Clients connected to the proxy initiate check in on the server and hop through to clients.

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.

Port number	Direction	Protocol	Notes
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.
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**

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 Uyuni을 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 향상하는 데 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. Install Uyuni Server

There are various scenarios to deploy a Uyuni Server.

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 유틸리티**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. Container Host General Requirements

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/

This procedure is described below.

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/ima
ges/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 유틸리티에 대한 자세한 내용은 Uyuni 컨테이너 유틸리티를 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 Overview

[command] **mgradm** 도구를 사용하여 Uyuni을(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.

Uyuni server hosts that are hardened for security may restrict execution of files from

the **/tmp** folder. In such cases, as a workaround, export the **TMPDIR** environment variable to another existing path before running **mgradm**.

예:

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

In Uyuni updates, tools will be changed to make this workaround unnecessary.

절차: Podman을 사용한 Uyuni 컨테이너 배포

1. 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"
```

2. 배포가 완료될 때까지 기다립니다.
3. 브라우저를 열고 서버 FQDN으로 이동합니다.

3.1.1.6.2. 영구 볼륨

많은 사용자가 영구 볼륨의 위치를 지정하기를 원할 것입니다.

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, 보안 수준이 높은 모든 환경에서 사용됩니다.

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

Procedure

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

+ .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
```

+ .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:

+ .Load the server image

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

3.1.2.1.1. Deploy Uyuni on openSUSE Tumbleweed

Uyuni also provides all the needed container images in RPM packages that can be installed on the system.

User should make the needed RPM available on the internal network. That can be done by using a second Uyuni Server or any kind of mirror.

Procedure: Install Uyuni on openSUSE Tumbleweed in Air-gapped

1. Install openSUSE Tumbleweed.
2. 시스템을 업데이트합니다.
3. Install tools packages and image packages (replace $ARCH$ with the correct architecture):

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

4. Deploy Uyuni with **mgradm**. In an Air-gapped environment you may want to use the option **--pullPolicy Never**.

For more detailed information about installing Uyuni Server on openSUSE Tumbleweed, see Server Deployment.

Uyuni 서버를 업그레이드하려면 시스템의 모든 패키지를 업그레이드하고 서버 업그레이드에 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 2025.10 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. Generate Proxy Configuration

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

For the containerized Uyuni Proxy, you must build a new proxy configuration file and then redeploy the container for the changes to take effect. This is the process for updating settings, including the SSL certificate.

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.

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.

3.2.1.1.2. Generate Proxy Configuration With spacecmd and Self-Signed Certificate

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. Generate Proxy Configuration With spacecmd and Custom Certificate

You can generate a Proxy configuration using **spacecmd** for custom certificates rather than the default self-signed certificates.

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. If your setup uses an intermediate CA, copy it as well and include it in the command with the **-i** option (can be provided multiple times if needed) :

```
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

Both **spacecmd** command and generating via Web UI ways create a configuration archive. This archive needs to be made available on container host. Transfer this generated archive to the container host.

3.2.1.3. Start Uyuni Proxy Containers

Container can be started with the **mgrpxy** command.

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
```

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

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

3.2.2. Uyuni Proxy Deployment on openSUSE Tumbleweed

이 가이드에서는 Uyuni 2025.10 프록시에 대한 배포 프로세스를 간략하게 설명합니다. 이 가이드에서는 Uyuni 2025.10 서버가 이미 배초된 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 구성을 생성합니다.
5. 프록시 구성을 서버에서 프록시로 전송합니다.
6. 프록시 구성을 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

Hardware	Details	Recommendation
CPU	x86-64, ARM	Minimum 2 dedicated 64-bit CPU cores
RAM	Minimum	2 GB
	Recommended	8 GB
Disk Space	/ (root directory)	Minimum 40 GB
	/var/lib/containers/storage/volumes	Minimum 100 GB, Storage requirements should be calculated for the number of ISO distribution images, containers, and bootstrap repositories you will use.

3.2.2.2. Container Host General Requirements

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. Log in.
3. Add the following repository to your openSUSE Tumbleweed server:

```
zypper ar
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/ima
ges/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 유틸리티를 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.

For more information, see

- **Installation-and-upgrade › Container-management**
- **Administration › Troubleshooting**

3.2.2.6. 프록시 호스트를 미니언으로 부트스트랩

작업: 프록시 호스트 부트스트랩

1. **시스템 › 부트스트랩**을 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 구성을 생성하기 전에 Uyuni 서버에 salt 미니언으로 등록해야 합니다.

다음 작업을 수행합니다.

Procedure:

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.

   서버에서 생성된 CA 인증서는 **/var/lib/containers/storage/volumes/root/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 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

1. 서버 컨테이너에서 서버 호스트 OS로 파일을 복사합니다.

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

2. 다음으로 서버 호스트 OS에서 프록시 호스트로 파일을 복사합니다.

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

3. 다음을 사용하여 프록시 설치:

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

3.2.2.9. Uyuni 2025.10 프록시 시작

이제 **mgrpxy** 명령으로 컨테이너를 시작할 수 있습니다.

작업: 프록시 시작 및 상태 확인

1. 다음을 호출하여 프록시 시작:

```
mgrpxy start
```

2. 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-httpds**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 구성을 전역 기본값으로 재설정한 다음 다시 로드합니다.

3.2.3. Proxy conversion from client

3.2.3.1. Overview

This chapter describes how to convert a client system into a Uyuni Proxy using the Web UI.

It assumes that the proxy host system has already been bootstrapped and subscribed to the base operating system channel.

For information about client onboarding, see **Client-configuration › Registration-overview**.

3.2.3.2. 요구사항

Before starting the conversion, ensure the following requirements are fulfilled.

3.2.3.2.1. Client Must Be

- Already onboarded in Uyuni
- Reachable via the network

3.2.3.3. Preparation

Before proceeding with the proxy conversion, make sure the following preparations are completed to avoid interruptions during the conversion process.

3.2.3.3.1. SSL Certificates

Valid SSL certificates are required to secure communication between the proxy and other components.

You need:

- The public certificate of the Certificate Authority (CA) that signed the certificate on the Uyuni server
- A certificate for the proxy.
- The corresponding private key for the proxy certificate.

If your CA uses an intermediate certificate chain, you must include all intermediate

certificates as well.

If you are not using third party certificates, you can generate them using the **rhn-ssl-tool** inside the Uyuni container.

Generate a proxy certificate

1. On the Uyuni server host, run:

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

   For more information about other parameters, see **Administration › Ssl-certs-selfsigned**.

2. Transfer the certificates to Uyuni server host

```
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
```

To confirm the exact folder where the certificates and key files were generated, you can list the directories with:

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

3. Transfer the certificates from Uyuni server host

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

3.2.3.3.2. Packages Preparation

Install mgrpxy

The **mgrpxy** tool must be installed from a repository matching your system. Choose the appropriate repository from:

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

목록 1. 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
```

Install Container Images

It is recommended to deploy the container images as RPM packages. Please ensure the following packages are installed on the client:

```
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
```

For details on air-gapped deployment, see **Installation-and-upgrade › Container-deployment**

3.2.3.4. Setup Proxy Client

1. Navigate to the client's **Overview** page.
2. Click button [**Convert to Proxy**].

   Confirm you were redirected to the proxy configuration form.

   This page can be accessed later from the **Details** > **Proxy** > **Configuration** tab.

3. In the Web UI, navigate to **Proxy › Configuration** and fill in the required data:

Procedure: Configuring the Proxy

a. In the **Parent FQDN** field, type the fully qualified domain name for the parent server or proxy.

b. In the **Proxy SSH port** field, type the SSH port on which the SSH service is listening on the Uyuni Proxy. It is recommended to keep the default: 8022.

c. In the **Max Squid cache size** field, type the maximum allowed size for the Squid cache, in Gigabytes.

d. In the **Proxy admin email** field, type the administrator's email address.

e. In the **Certificates** section, provide the certificates for the Uyuni Proxy, obtained in the preparation step.

f. In the **Source** section, select one of the two options: **RPM** or **Registry**.

- The **RPM** option is recommended for air-gapped or restricted environments. 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**.
  - If **Simple** is selected, provide values in the **Registry URL** and **Containers Tag** fields.
    - For **Registry URL** use: **registry.opensuse.org/uyuni**.
    - Select the tag from the drop-down list.
  - If **Advanced** is selected, an additional section of the form is shown:

- For each individual container URL field, use the registry: **registry.opensuse.org/uyuni** followed by the corresponding suffix, for example, **proxy-httpd** or **salt-broker**.
- Select the tag from the drop-down list.

4. Once all fields are filled, click **[Apply]** to apply the configuration and schedule the proxy installation task.

3.2.3.5. Verify Proxy Activation

Check the client's event history to confirm task success.

(Optional) Access the proxy's HTTP endpoint to validate it shows a welcome page.

3.2.4. Uyuni Proxy Deployment on K3s

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.

Install Helm by using the installer script:

```
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
```

For more information, see https://helm.sh/docs/intro/install/#from-script.

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

1. To install package on Leap Micro run:

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

2. 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-deployment에서의 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) documentation

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: 2025.10).
- 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/ima
ges/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: Prepareing 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. Legacy Uyuni Server Migration to Container

To migrate a legacy Uyuni Server to a container, a new machine is required.

In the context of this migration, the legacy Uyuni Server (RPM installation) is sometimes also called old server.

4.1.2.1. Requirements and Considerations

4.1.2.1.1. Hostnames

Neither in-place migration is not possible nor allows the migration procedure currently any hostname renaming functionality.

Thus the fully qualified domain name (FQDN) on the new server will remain identical to that on the legacy server.

After migration, it is necessary to update the DHCP and DNS records to point to the new server.

For more information, see Finalize migration.

4.1.2.1.2. SSL certificates

SSL certificates are needed at a later stage. If not using the self-signed generated CA and certificates, ensure you have the following before starting:

- A certificate authority (CA) SSL public certificate. If you are using a CA chain, all intermediate CAs must also be available.
- An SSL database private key.
- An SSL database certificate.

All files must be in PEM format.

The hostname of the SSL server certificate must match the fully qualified hostname of the machine you deploy them on. You can set the hostnames in the **X509v3 Subject Alternative Name** section of the certificate. You can also list multiple hostnames if your environment requires it. Supported Key types are **RSA** and **EC** (Elliptic Curve).

Database SSL certificate requires **reportdb** and **db** and the FQDN used to access the report database as **Subject Alternative Name**.

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 keys

- 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.
- The administrator must migrate these keys manually from the legacy Uyuni installation to the container host after the actual server migration.

Procedure: Manual Migration of the GPG Keys to New Server

1. Copy the keys from the legacy Uyuni server to the container host of the new server.
2. 그 후, **mgradm gpg add <PATH_TO_KEY_FILE>** 명령을 사용하여 마이그레이션된 서버에 각 키를 추가합니다.

4.1.2.2.1. Initial Preparation on the Legacy Server

The migration can take a very long time depending on the amount of data that needs to be replicated. To reduce downtime it is possible to run the migration multiple times in a process of initial replication, re-replication, or final replication and switch over while all the services on the legacy server can stay up and running.

Only during the final migration the processes on the legacy server need to be stopped.

For all non-final replications add the parameter **--prepare** to prevent the automatic stopping the services on the legacy server. For example:

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

Procedure: Initial Preparation on the Legacy Server

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. Ensure that for **root** an SSH key exists on the new 2025.10 server. If a key does not exist, create it with the command:

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

2. The SSH configuration and agent should be ready on the new server host for a connection to the legacy server that does not prompt for a password.

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

To establish a connection without prompting for a password, the migration script relies on an SSH agent running on the new server. If the agent is not active yet, initiate it by running **eval $(ssh-agent)**. Then add the SSH key to the running agent with **ssh-add** followed by the path to the private key. You will be prompted to enter the password for the private key during this process.

3. Copy the public SSH key to the legacy Uyuni Server (**<oldserver.fqdn>**) with **ssh-copy-id**. Replace **<oldserver.fqdn>** with the FQDN of the legacy server:

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

The SSH key will be copied into the legacy server's **~/.ssh/authorized_keys** file. For more information, see the **ssh-copy-id** manpage.

4. Establish an SSH connection from the new server to the legacy Uyuni Server to check that no password is needed. Also there must not by any problem with the host fingerprint. In case of trouble, remove old fingerprints from the **~/.ssh/known_hosts** file. Then try again. The fingerprint will be stored in the local **~/.ssh/known_hosts** file.

4.1.2.2.3. 마이그레이션 수행

When planning your migration from a legacy Uyuni to a containerized Uyuni, ensure that your target instance meets or exceeds the specifications of the legacy setup. This includes, but is not limited to, memory (RAM), CPU Cores, Storage, and Network Bandwidth.

Uyuni server hosts that are hardened for security may restrict execution of files from the **/tmp** folder. In such cases, as a workaround, export the **TMPDIR** environment variable to another existing path before running **mgradm**.

예:

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

In Uyuni updates, tools will be changed to make this workaround unnecessary.

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>]
```

Devices must not have any filesystem. The command aborts if a filesystem exists on the storage device.

예:

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

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

For more information, see

- **Installation-and-upgrade › Container-management**
- **Administration › Troubleshooting**

마이그레이션 수행

1. Execute the following command to install a new Uyuni server. Replace **<oldserver.fqdn>** with the FQDN of the legacy server:

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

2. 신뢰할 수 있는 SSL CA 인증서를 마이그레이션합니다.

Migration of the Certificates

Trusted SSL CA certificates that were installed as part of an RPM and stored on a legacy Uyuni in the **/usr/share/pki/trust/anchors/** directory will not be migrated. Because SUSE does not install RPM packages in the container, the administrator must migrate these certificate files manually from the legacy installation after migration:

Procedure: Migrating the Certificates

1. Copy the file from the legacy server to the new server. 예를 들어 **/local/ca.file**일 수 있습니다.
2. Copy the file into the container with the command:

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

Finalize migration

After successfully running the **mgradm migrate** command, the Salt setup on all clients will still point to the legacy server.

To redirect them to the new 2025.10 server, it is required to rename the new server at the infrastructure level (DHCP and DNS) to use the same FQDN and IP address as the legacy server.

If something goes wrong with the migration it is possible to restart the old system. As root, restart PostgreSQL and the spacewalk services with the following commands:

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

4.1.2.3. Kubernetes Preparations

Before executing the migration with **mgradm migrate** command, it is essential to predefine **Persistent Volumes**, especially considering that the migration job initiates the container from scratch.

For more information, see the installation section on preparing these volumes in **Installation-and-upgrade › Container-management**.

4.1.2.4. 마이그레이션

Execute the following command to install a new Uyuni server, replacing **<oldserver.fqdn>** with the appropriate FQDN of the legacy server:

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

또는

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

After successfully running the **mgradm migrate** command, the Salt setup on all clients will still point to the legacy server. To redirect them to the new server, it is required to rename the new server at the infrastructure level (DHCP and DNS) to use the same FQDN and IP address as the legacy server.

4.1.3. Uyuni Server Upgrade

Before running the upgrade command, it is required to update the host operating system. Updating the host operating system will also result in the update of the Uyuni tooling such as the **mgradm** tool.

Procedure: Upgrading Server

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.

Upgrading with third-party SSL certificate

If you are using third-party certificates, the database container needs to have an SSL certificate with the following Subject Alternate Names (SANs):

- **db**
- **reportdb**
- the externally facing fully qualified domain name

The same certificate can be used for both the main container and the database one, but it needs to have those SANs too.

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. Database Backup Volume

Server migration or upgrade with **mgradm migration** or **mgradm upgrade** can create a volume with the database backup.

When the PostgreSQL database version is increased, the old database must be stored in a separate location before running the upgrade. For this purpose **mgradm** dynamically creates the volume **var-pgsql-backup**. When the migration or upgrade is done and the user has validated that the new system is working as expected, this volume can be removed safely.

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 **mgrpxy**.

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\([^:]\+\):.*/\1/'
systemctl cat uyuni-proxy-squid.service | grep EXTRA_CONF= | sed 's/.*=-
v\([^:]\+\):.*/\1/'
```

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.

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/ima
ges/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/
```

2. 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
```

3. 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

1. Check containers are running:

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

2. 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. Legacy Proxy Migration to Container

The containerized proxy now is managed by a set of systemd services. For managing the containerized proxy, use the **mgrpxy** tool.

This section will help you migrate from the legacy **systemd** proxy using the **mgrpxy** tool.

An in-place migration from previous releases of Uyuni to 2025.10 will remain unsupported due to the HostOS change from openSUSE Leap to openSUSE Leap Micro.

The traditional contact protocol is no longer supported in Uyuni 2025.10 and later. Before migrating from previous Uyuni releases to 2025.10, any existing traditional clients including the traditional proxies must be migrated to Salt.

4.2.2.1. Migrate From Legacy to Containerized Proxy With Systemd

4.2.2.1.1. Generate Proxy Configuration

Procedure: Generate the Proxy Configuration

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 구성을 새 호스트로 전송

Procedure: Transfering the Proxy Configuration

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-httpd
proxy-tftpd
proxy-squid
proxy-ssh
```

4.2.2.2. Migrate Uyuni Proxy to Uyuni 2025.10 Containerized Proxy

Procedure: Migrate Uyuni Containerized Proxy to Uyuni 2025.10 New Containerized Proxy

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. Copy your **tar.gz** proxy configuration to the host.

4.2.2.3. Install Packages Using the Web UI

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

Procedure: Installing Packages Using the Web UI

1. After installation, ensure that the SLE Micro 6.1 parent channel and Proxy child channels are added and synchronized from the **Admin › Setup Wizard → Products** page.
2. In the Web UI, go to **Systems › Activation Keys** and create an activation key linked for the synchronized SLE Micro 6.1 channel.
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. **mgrpxy** 및 **mgrpxy-bash-completion** 패키지를 설치합니다.
6. 시스템을 재부팅합니다.

4.2.2.4. Generate Proxy Config With spacecmd and Self-Signed Certificate

spacecmd를 사용하여 프록시 구성을 생성할 수 있습니다.

Procedure: Generate Proxy Config With spacecmd and Self-Signed Certificate

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. Generate Proxy Config With spacecmd and Custom Certificate

You can generate Proxy configuration using **spacecmd** for a custom certificates rather than default self-signed certificates.

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

Procedure: Generate Proxy Config With spacecmd and Custom Certificate

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 Proxy Upgrade

Before running the upgrade command, it is required to update the host operating system. Updating the host operating system will also result in the update of the Uyuni tooling such as the **mgrpxy** tool.

Procedure: Upgrading Proxy

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 Proxy containers running on **podman** can be updated using the following command:

```
mgrpxy upgrade podman
```

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

```
mgrpxy upgrade kubernetes
```

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.

We highly recommend using the same tag for all proxy containers to ensure consistency under normal circumstances.

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

4.3. 클라이언트

4.3.1. Upgrade Clients

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. mgradm을 사용하여 사용자 지정 YAML 구성 및 배포

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 2025.10 서버 컨테이너를 재시작, 시작 및 중지할 수 있습니다.

Uyuni 2025.10 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 2025.10.

표 10. Server Containers

Container Name	Description
uyuni-server	Main product container
uyuni-db	Database container for the product
uyuni-hub-xmlrpc	XML-RPC gateway for Hub deployment
uyuni-server-attestation	Server COCO attestation
uyuni-saline	Saline container for Salt observability
uyuni-server-migration	Migration helper container

표 11. Proxy Containers

Container Name	Description
uyuni-proxy-httpd	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. Persistent Container Volumes

컨테이너 내에서 수행한 수정 사항은 유지되지 않습니다. 영구 볼륨 외부에서 변경한 내용은 모두 삭제됩니다. 아래는 Uyuni 2025.10에 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/

Volume Name	Volume Directory
var-cobbler	**/var/lib/cobbler**
var-salt	**/var/lib/salt**
var-pgsql	**/var/lib/pgsql/data**
var-pgsql-backup	**/var/lib/pgsql-backup**

Volume Name	Volume Directory
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. Persistent Volumes: run/

Volume Name	Volume Directory
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. Understanding mgr-storage-server and mgr-storage-proxy

mgr-storage-server and **mgr-storage-proxy** are helper scripts provided with Uyuni.

They are designed to configure storage for Uyuni Server and Proxy.

The scripts take disk devices as arguments. **mgr-storage-proxy** requires a single argument for the storage disk device. **mgr-storage-server** requires a storage disk device and can optionally accept a second argument for a dedicated database disk device. While both normal and database storage can reside on the same disk, it is advisable to place the database on a dedicated, high-performance disk to ensure better performance and easier management.

5.5.1. What these tools do

Both **mgr-storage-server** and **mgr-storage-proxy** perform standard storage setup operations:

- Validate the provided storage devices.
- Ensure that devices are empty and suitable for use.
- Create XFS filesystems on the specified devices.
- Mount the devices temporarily for data migration.

- Move the relevant storage directories to the new devices.
- Create entries in **/etc/fstab** so that the storage mounts automatically on boot.
- Remount the devices at their final locations.

표 21. Additional tool-specific behavior

mgr-storage-server	• Optionally supports a separate device for database storage. • Stops SUSE Manager services during migration, restarts them afterward. Moves Podman volumes directory **/var/lib/containers/storage/volumes** to the prepared storage, and optionally **/var/lib/containers/storage/volumes/var-pgsql** to the prepared database storage.
mgr-storage-proxy	• Focuses only on proxy storage (no database storage support). • Stops and restarts the proxy service during migration. • Moves podman volumes directory **/var/lib/containers/storage/volumes** to the prepared storage.

Both tools automate standard Linux storage operations. There is no hidden or custom logic beyond what a Linux administrator would do manually.

5.5.2. What these tools do not do

- They do **not** create or manage LVM volumes.
- They do **not** configure RAID or complex storage topologies.
- They do **not** prevent you from managing storage using normal Linux tools after setup.
- They do **not** provide dynamic resizing or expansion capabilities — these must be handled using standard Linux storage tools.

5.5.3. Post-installation storage management

Once storage has been configured, you can safely manage it using standard Linux commands.

5.5.3.1. Examples

목록 2. Example 1: Extending storage if using LVM

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

Example 2: Migrating to a larger disk

1. Add and format the new disk.
2. Mount it temporarily.
3. Use **rsync** to copy data.

4. Update **/etc/fstab**.
5. Remount at the correct location.

5.5.4. When to use, or not use

Always take a backup before making changes to your storage setup.

- Use these tools **only** during initial storage setup or when migrating to new storage where the tool is expected to handle data migration and update **/etc/fstab**.
- Do **not** rerun these scripts for resizing or expanding storage. Use standard Linux tools (e.g., **lvextend**, **xfs_growfs**) for such operations.

5.5.5. Summary

mgr-storage-server and **mgr-storage-proxy** help automate the initial persistent storage setup for Uyuni components using standard Linux storage practices. They do not limit or interfere with standard storage management afterward.

After setup, continue managing your storage using familiar Linux tools.

A full database volume can cause significant issues with system operation. As disk usage notifications have not yet been adapted for containerized environments, users are encouraged to monitor the disk space used by Podman volumes themselves, either through tools such as Grafana, Prometheus, or any other preferred method. Pay particular attention to the var-pgsql volume, located under **/var/lib/containers/storage/volumes/**.

장 6. 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".