

Uyuni 2026.01

インストールおよびアップ グレードガイド



U Y U N I

章 1. 序文

Installation, Deployment and Upgrade + Uyuni 2026.01

このガイドでは、Uyuniサーバおよびプロキシを配備、アップグレード、管理するための包括的な手順を段階的に説明します。

以下のセクションで構成されています。

- **要件:** スムーズなセットアップを確実に行うために必要なハードウェア、ソフトウェア、およびネットワークの前提条件を概説します。
 - **配備とインストール:** Uyuniをコンテナとして配備し、初期設定を完了する手順を説明します。
 - **アップグレードと移行:** ダウンタイムを最小限に抑えながらUyuniをアップグレードおよび移行するプロセスを詳細に説明します。
 - **基本的なサーバ管理:** 基本的なサーバ操作をカバーし、Uyuniを効率的に開始するのに役立ちます。

発行日: 2026-01-28

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

目次

| | |
|--|----|
| 1. 序文 | 1 |
| 2. 要件 | 4 |
| 2.1. 一般的な要件 | 4 |
| 2.1.1. サーバ要件 | 4 |
| 2.1.2. プロキシ要件 | 4 |
| 2.2. ネットワーク要件 | 5 |
| 2.2.1. 完全修飾ドメイン名(FQDN) | 5 |
| 2.2.2. ホスト名とIPアドレス | 6 |
| 2.2.3. ルータ広告の再有効化 | 6 |
| 2.2.4. HTTPまたはHTTPSのOSIレベル7プロキシの背後での配備 | 7 |
| 2.2.5. Air-gapped配備 | 8 |
| 2.2.6. 必須のネットワークポート | 8 |
| 2.3. パブリッククラウドの要件 | 15 |
| 2.3.1. ネットワーク要件 | 16 |
| 2.3.2. ストレージボリュームの準備 | 16 |
| 3. 配備とインストール | 18 |
| 3.1. Uyuniサーバのインストール | 18 |
| 3.1.1. Uyuni Server Deployment on openSUSE Tumbleweed | 18 |
| 3.1.2. UyuniサーバのAir-gapped配備 | 21 |
| 3.2. Uyuniプロキシのインストール | 23 |
| 3.2.1. コンテナ化されたUyuniプロキシの設定 | 23 |
| 3.2.2. Uyuni Proxy Deployment on openSUSE Tumbleweed | 26 |
| 3.2.3. クライアントからのプロキシ変換 | 32 |
| 3.2.4. K3sへのUyuniプロキシの配備 | 35 |
| 4. アップグレードと移行 | 37 |
| 4.1. サーバ | 37 |
| 4.1.1. Migrating the Uyuni Server to openSUSE Tumbleweed | 37 |
| 4.1.2. レガシUyuniサーバからコンテナへの移行 | 40 |
| 4.1.3. Uyuniサーバのアップグレード | 44 |
| 4.2. プロキシ | 46 |
| 4.2.1. Migrating the Uyuni Proxy to openSUSE Tumbleweed | 46 |
| 4.2.2. レガシプロキシからコンテナへの移行 | 49 |
| 4.2.3. Uyuniプロキシのアップグレード | 53 |
| 4.3. クライアント | 54 |
| 4.3.1. クライアントのアップグレード | 54 |
| 5. 基本的なサーバとプロキシの管理 | 55 |
| 5.1. Custom YAML Configuration and Deployment with mgradm | 55 |
| 5.2. コンテナの起動と停止 | 56 |
| 5.3. Uyuniで使用されるコンテナ | 56 |
| 5.4. 永続コンテナボリューム | 57 |
| 5.4.1. サーバ | 57 |
| 5.4.2. プロキシ | 59 |
| 5.5. mgr-storage-server および mgr-storage-proxy について | 59 |
| 5.5.1. これらのツールの機能 | 60 |
| 5.5.2. これらのツールが実行しないこと | 60 |
| 5.5.3. インストール後のストレージ管理 | 60 |
| 5.5.4. 使用する場合、使用しない場合 | 61 |
| 5.5.5. 概要 | 61 |
| 6. GNU Free Documentation License | 62 |

章 2. 要件

2.1. 一般的な要件

次の表では、サーバとプロキシの最低要件を指定しています。



- ■ ■ NFSはSELinuxのファイルラベル付けをサポートしていないため、NFSをストレージに使用しないでください。

2.1.1. サーバ要件

表 1. x86-64アーキテクチャのサーバ要件

| 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)

* 各SUSE製品およびPackage Hubでは50GB

各Red Hat製品では360 GB |
| | /var/cache | 10 GB以上。 SUSE製品あたり100 MB、Red Hatまたは他の製品あたり1 GBを追加します。 サーバがISSマスタである場合は容量を倍増します。 |
| | スワップ容量 | 3 GB |

2.1.2. プロキシ要件

表 2. プロキシ要件

| Software and Hardware | Details | Recommendation |
|-----------------------|--------------------------------|--------------------------------------|
| Tumbleweed | Clean installation, up-to-date | Tumbleweed |
| 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プロキシは、**/var/cache**ディレクトリにパッケージをキャッシュします。**/var/cache**の容量が不足している場合、プロキシは、使用されていない古いパッケージを削除し、新しいパッケージに置き換えます。

この動作の結果は以下のとおりです。

- プロキシ上の**/var/cache**ディレクトリの容量を大きくすると、このディレクトリとUyuniサーバ間のトラフィックが少なくなります。
- プロキシ上の**/var/cache**ディレクトリをUyuniサーバ上の**/var/spacewalk**と同じサイズにすることで、最初に同期した後のトラフィック量の増大が防止されます。
- /var/cache**ディレクトリは、Uyuniサーバ上ではプロキシと比べて小さくできます。サイズの推測のガイドについては、[\[server-hardware-requirements\]](#)セクションを参照してください。

2.2. ネットワーク要件

このセクションでは、Uyuniのネットワークとポートの要件について詳しく説明します。

IP転送は、コンテナ化されたインストールによって有効になります。つまり、Uyuniサーバとプロキシはルータとして動作します。この動作はpodmanによって直接実行されます。IP転送が無効になっている場合、Podmanコンテナは実行されません。



ポリシーに従って、Uyuni環境のネットワーク分離を実現することを検討してください。

詳細については、<https://www.suse.com/support/kb/doc/?id=000020166>を参照してください。

2.2.1. 完全修飾ドメイン名(FQDN)

Uyuniサーバは、そのFQDNを正しく解決する必要があります。FQDNを解決できない場合、多数のコンポーネントで重大な問題の原因になる場合があります。

ホスト名とDNSの設定の詳細については、<https://documentation.suse.com/sles/15-SP6/html/SLES-all/cha-network.html#sec-network-yast-change-host>を参照してください。

2.2.2. ホスト名とIPアドレス

Uyuniのドメイン名をそのクライアントで解決できることを確認するには、サーバとクライアントの両方のマシンを動作中のDNSサーバに接続する必要があります。リバース参照が正しく設定されていることも確認する必要があります。

DNSサーバの設定の詳細については、<https://documentation.suse.com/sles/15-SP6/html/SLES-all/cha-dns.html>を参照してください。

2.2.3. ルータ広告の再有効化

Uyuniをmgradm install podmanまたはmgrpxy install podmanを使用してインストールすると、IPv4とIPv6の転送を有効にするPodmanが設定されます。これは、コンテナの外側から通信を行うために必要です。

ただし、システムで以前に/**proc/sys/net/ipv6/conf/eth0/accept_ra**を1に設定していた場合、ルータ広告の使用が停止されます。その結果、ルートがルータ広告経由で取得されなくなり、デフォルトのIPv6ルートが消失します。

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

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

Procedure: Reenabling router advertisements without Network Manager

1. /etc/sysctl.dにファイルを作成します。例: **99-ipv6-ras.conf**。
2. 次のパラメータと値をファイルに追加します。

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

3. 再起動します。

Procedure: Reenabling router advertisements with Network Manager

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

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

3. 再起動します。
4. The file should look similar to this:

```
[connection]
id=Wired connection 1
type=ethernet
interface-name=eth0

[ethernet]

[ipv4]
dns-priority=20
method=auto

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

2.2.4. HTTPまたはHTTPSのOSIレベル7プロキシの背後での配備

一部の環境では、HTTPまたはHTTPSプロキシを介したインターネットアクセスが強制されています。 これはSquidサーバなどのサーバが対象となります。 このような設定でUyuniサーバのインターネットアクセスを許可するには、次のように設定する必要があります。

プロシージャ: HTTPまたはHTTPS OSIレベル7プロキシの設定

1. オペレーティングシステムのインターネットアクセスの場合は、必要に応じて`/etc/sysconfig/proxy`を変更します。

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

2. Podmanコンテナのインターネットアクセスの場合は、必要に応じて`/etc/systemd/system/uyuni-server.service.d/custom.conf`を変更します。たとえば、次のように設定します。

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

3. Javaアプリケーションのインターネットアクセスの場合は、必要に応じて`/etc/rhn/rhn.conf`を変更します。コンテナホストで、`mgrctl term`を実行し、サーバコンテナ内でコマンドラインを開きます。

- a. 必要に応じて `/etc/rhn/rhn.conf` を変更します。たとえば、次のように設定します。

```
# 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 separated list
server.satellite.no_proxy =
```

4. コンテナホストでサーバを再起動して、新しい設定を適用します。

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

2.2.5. Air-gapped配備

社内ネットワーク上で操作していて、SUSE Customer Centerにアクセスできない場合、**Installation-and-upgrade** › **Container-deployment**を使用できます。

運用環境では、Uyuniサーバおよびクライアントはファイアウォールを常に使用する必要があります。必要なポートの一覧は、[installation-and-upgrade:network-requirements.pdf](#)を参照してください。

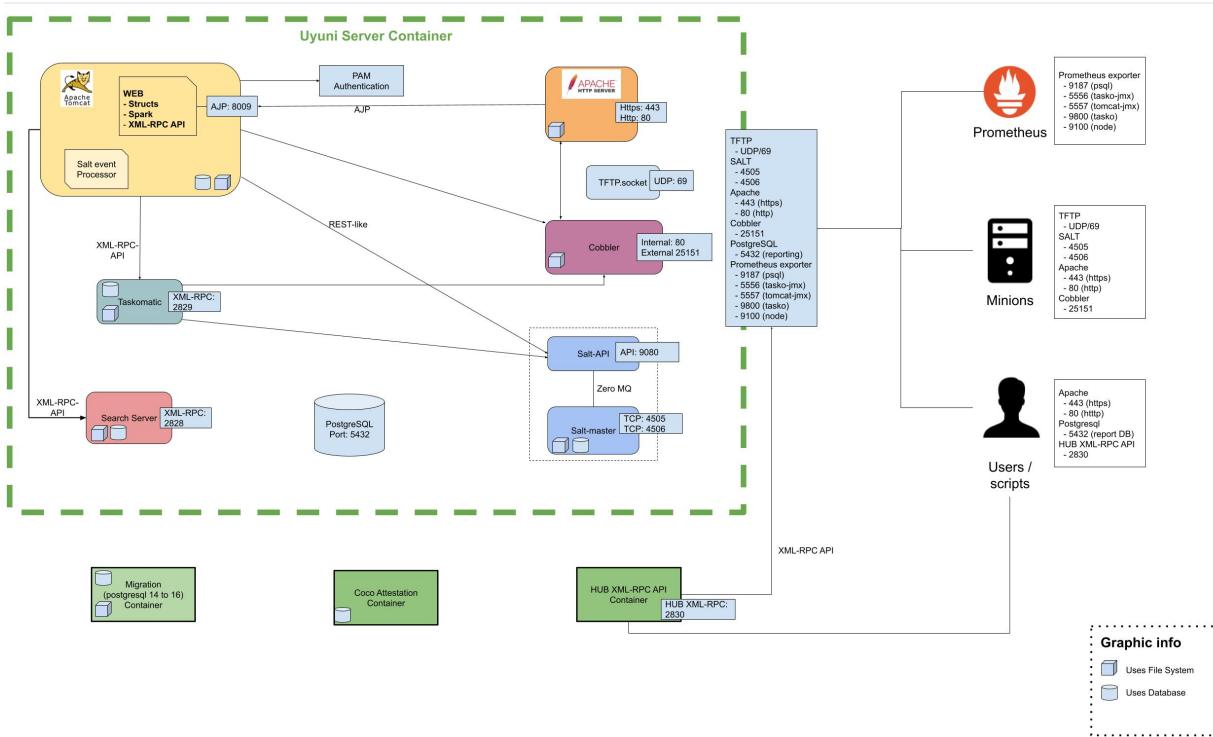
2.2.6. 必須のネットワークポート

このセクションには、Uyuni内でさまざまな通信に使用するポートの一覧が記載されています。

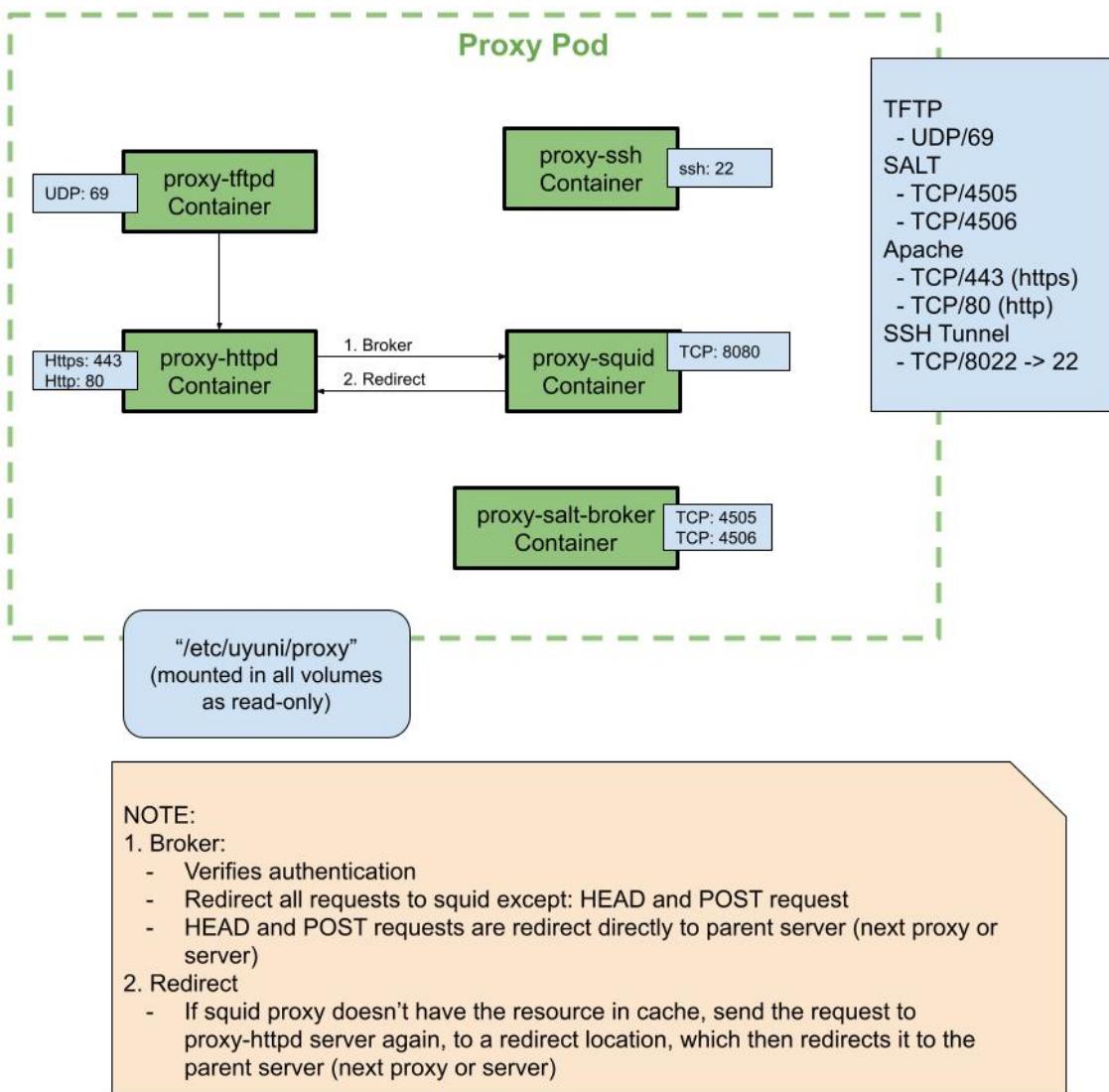
これらのポートすべてを開く必要はありません。サービスの使用に必要なポートのみを開く必要があります。

2.2.6.1. 概要

2.2.6.1.1. サーバ



2.2.6.1.2. プロキシ



2.2.6.2. 外部の着信サーバポート

未許可アクセスからサーバを保護するためにUyuniサーバでファイアウォールを設定するには、外部の着信ポートが開いている必要があります。

これらのポートを開くと、外部ネットワークトラフィックがUyuniサーバにアクセスできるようになります。

表 3. Uyuniサーバの外部ポートの要件

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|---------|------|---------------------------------------|
| 67 | TCP/UDP | DHCP | クライアントがサーバからIPアドレスをリクエストしている場合のみ必要です。 |

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|---------|------------|---|
| 69 | TCP/UDP | TFTP | 自動化されたクライアントのインストールのためにサーバがPXEサーバとして使用されている場合に必要です。 |
| 80 | TCP | HTTP | 一部のブートストラップリポジトリおよび自動化されたインストールのために一時的に必要です。 |
| 443 | TCP | HTTPS | Web UI、クライアント、およびサーバとプロキシ(tftpsync)のリクエストを処理します。 |
| 4505 | TCP | salt | クライアントからの通信リクエストを受け入れるために必要です。クライアントは、接続を開始し、開いたままになり、Saltマスタからのコマンドを受信します。 |
| 4506 | TCP | salt | クライアントからの通信リクエストを受け入れるために必要です。クライアントは、接続を開始し、開いたままになり、Saltマスタに結果を返します。 |
| 5432 | TCP | PostgreSQL | レポーティングデータベースにアクセスするために必要です。 |
| 5556 | TCP | Prometheus | Taskomatic JMXのメトリックをスクレイピングするために必要です。 |
| 5557 | TCP | Prometheus | Tomcat JMXのメトリックをスクレイピングするために必要です。 |
| 9100 | TCP | Prometheus | Node exporterのメトリックをスクレイピングするために必要です。 |
| 9187 | TCP | Prometheus | PostgreSQLのメトリックをスクレイピングするために必要です。 |
| 9800 | TCP | Prometheus | Taskomaticのメトリックをスクレイピングするために必要です。 |
| 25151 | TCP | Cobbler | |

2.2.6.3. 外部の送信サーバポート

サーバからアクセスできるアクセス先を制限するためにUyuniサーバでファイアウォールを設定するには、外部の送信ポートが開いている必要があります。

次のポートを開くと、Uyuniサーバからのネットワークトラフィックで外部サービスに通信できます。

表 4. Uyuniサーバの外部ポートの要件

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|-------|---------|---|
| 80 | TCP | HTTP | SUSE Customer Centerで必要です。ポート80はWeb UIを操作するためには使用されません。 |
| 443 | TCP | HTTPS | SUSE Customer Centerで必要です。 |
| 25151 | TCP | Cobbler | |

2.2.6.4. 内部サーバポート

内部ポートは、Uyuniサーバによって内部で使用されます。 内部ポートは**localhost**のみからアクセスできます。

ほとんどの場合、これらのポートを調整する必要はありません。

表 5. Uyuniサーバの内部ポートの要件

| ポート番号 | 注意 |
|-------|---|
| 2828 | サテライト検索APIであり、TomcatとTaskomaticのRHNアプリケーションで使用されます。 |
| 2829 | Taskomatic APIであり、TomcatのRHNアプリケーションで使用されます。 |
| 8005 | Tomcatのシャットダウンポート。 |
| 8009 | TomcatからApache HTTPD (AJP)。 |
| 8080 | TomcatからApache HTTPD (HTTP)。 |
| 9080 | Salt-APIであり、TomcatとTaskomaticのRHNアプリケーションで使用されます。 |
| 25151 | CobblerのXMLRPC API |
| 32000 | Taskomaticおよびサテライト検索を実行する仮想マシン(JVM)へのTCP接続用のポート。 |

ポート32768以上は一時ポートとして使用されます。これらは、TCP接続の受信に最も頻繁に使用されます。TCP接続リクエストが受信されると、送信元はこれらの一時ポート番号のいずれかを選択して、宛先ポートと照合します。

次のコマンドを使用して、一時ポートであるポートを確認できます。

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

2.2.6.5. 外部の着信プロキシポート

未許可アクセスからプロキシを保護するためにUyuniプロキシでファイアウォールを設定するには、外部の着信ポートが開いている必要があります。

これらのポートを開くと、外部ネットワークトラフィックがUyuniプロキシにアクセスできるようになります。

表 6. Uyuniプロキシの外部ポートの要件

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|---------|-------|---|
| 22 | | | ユーザがSalt SSHを使用してプロキシホストを管理する場合のみ必要です。 |
| 67 | TCP/UDP | DHCP | クライアントがサーバからIPアドレスをリクエストしている場合のみ必要です。 |
| 69 | TCP/UDP | TFTP | 自動化されたクライアントのインストールのためにサーバがPXEサーバとして使用されている場合に必要です。 |
| 443 | TCP | HTTPS | Web UI、クライアント、およびサーバとプロキシ(tftpsync)のリクエストを処理します。 |
| 4505 | TCP | salt | クライアントからの通信リクエストを受け入れるために必要です。クライアントは、接続を開始し、開いたままになり、Saltマスターからのコマンドを受信します。 |
| 4506 | TCP | salt | クライアントからの通信リクエストを受け入れるために必要です。クライアントは、接続を開始し、開いたままになり、Saltマスターに結果を返します。 |
| 8022 | | | ssh-pushおよびssh-push-tunnelの接続メソッドに必要です。プロキシに接続されているクライアントは、サーバへのチェックインを開始し、クライアントにホップします。 |

2.2.6.6. 外部の送信プロキシポート

プロキシからアクセスできるアクセス先を制限するためにUyuniプロキシでファイアウォールを設定するには、外部の送信ポートが開いている必要があります。

次のポートを開くと、Uyuniプロキシからのネットワークトラフィックで外部サービスに通信できます。

表 7. Uyuniプロキシの外部ポートの要件

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|-------|-------|-----------------------------------|
| 80 | | | サーバにアクセスするために使用します。 |
| 443 | TCP | HTTPS | SUSE Customer Centerで必要です。 |
| 4505 | TCP | Salt | Saltマスターに直接またはプロキシ経由で接続する必要があります。 |

| ポート番号 | プロトコル | 使用元 | 注意 |
|-------|-------|------|-----------------------------------|
| 4506 | TCP | Salt | Saltマスターに直接またはプロキシ経由で接続する必要があります。 |

2.2.6.7. 外部クライアントポート

Uyuniサーバとそのクライアントの間でファイアウォールを設定するには、外部クライアントポートが開いている必要があります。

ほとんどの場合、これらのポートを調整する必要はありません。

表 8. Uyuniクライアントの外部ポートの要件

| ポート番号 | 方向 | プロトコル | 注意 |
|-------|----|-------|---|
| 22 | 着信 | SSH | ssh-pushおよびssh-push-tunnelの接続メソッドに必要です。 |
| 80 | 送信 | | サーバまたはプロキシにアクセスするために使用します。 |
| 443 | 送信 | | サーバまたはプロキシにアクセスするために使用します。 |
| 4505 | 送信 | TCP | Saltマスターに直接またはプロキシ経由で接続する必要があります。 |
| 4506 | 送信 | TCP | Saltマスターに直接またはプロキシ経由で接続する必要があります。 |
| 9090 | 送信 | TCP | Prometheusユーザインターフェースに必要です。 |
| 9093 | 送信 | TCP | Prometheus警告マネージャに必要です。 |
| 9100 | 送信 | TCP | Prometheusノードエクスポートに必要です。 |
| 9117 | 送信 | TCP | Prometheus Apacheエクスポートに必要です。 |
| 9187 | 送信 | TCP | Prometheus PostgreSQLに必要です。 |

2.2.6.8. 必要なURL

クライアントを登録して更新を実行するためにUyuniがアクセスできる必要があるURLがあります。ほとんどの場合、次のURLにアクセスできれば十分です。

- scc.suse.com
- updates.suse.com
- installer-updates.suse.com
- registry.suse.com

- [registry-storage.suse.com](#)
- [opensuse.org](#)

さらに、SUSE以外の製品の場合は、次のURLへのアクセスが必要になる場合があります。

- [download.nvidia.com](#)
- [public.dhe.ibm.com](#)
- [nu.novell.com](#)

指定したURLとそれに関連するIPアドレスをホワイトリストに登録する方法の詳細については、次の記事を参照してください: [ファイアウォール内および/またはプロキシ経由でSUSE Customer CenterとSUSEレジストリにアクセスする](#)。

SUSE以外のクライアントを使用している場合、該当するオペレーティングシステム用の特定のパッケージを提供するその他のサーバにもアクセスできる必要がある場合があります。たとえば、Ubuntuクライアントがある場合、Ubuntuサーバにアクセスできる必要があります。

SUSE以外のクライアントでファイアウォールアクセスのトラブルシューティングを行う方法の詳細については、[Administration > Troubleshooting](#)を参照してください。

2.3. パブリッククラウドの要件

このセクションは、パブリッククラウドインフラストラクチャにUyuniをインストールする要件について説明します。Amazon EC2、Google Compute Engine、およびMicrosoft Azureではテストを実施済みですが、若干の差異はあってもその他のプロバイダにも当てはまるはずです。

始める前に、考慮事項を次に示します。

- Uyuni設定プロシージャは、正引きで確認された逆引きDNS参照を実行します。設定プロシージャが完了してUyuniが期待どおりに動作するためには、この参照が成功する必要があります。Uyuniを設定する前に、ホスト名とIPの設定を実行することが重要です。
- Uyuniサーバとプロキシのインスタンスは、DNSエントリーを介した制御を提供するネットワーク設定で実行する必要がありますが、大規模インターネットからはアクセスできません。
- このネットワーク設定内では、DNSの解決を提供する必要があります。**hostname -f**は、完全修飾ドメイン名(FQDN)を返す必要があります。
- DNSの解決は、クライアントを接続するためにも重要です。
- DNSは、選択したクラウドフレームワークに依存しています。詳細な手順については、クラウドプロバイダのドキュメントを参照してください。
- ソフトウェアリポジトリ、サーバデータベース、およびプロキシsquidキャッシュは外部仮想ディスクに配置することをお勧めします。こうすることによって、インスタンスが予期せずに終了した場合のデータ損失が防止されます。このセクションでは、外部仮想ディスクの設定方法の手順について説明します。

2.3.1. ネットワーク要件

パブリッククラウドでUyuniを使用する場合は、制限付きのネットワークを使用する必要があります。適切なファイアウォール設定でVPCプライベートサブネットを使用することをお勧めします。指定したIP範囲にあるマシンのみがインスタンスにアクセスできる必要があります。



パブリッククラウド上でUyuniを実行するということは、堅牢なセキュリティ対策を実装することを意味します。インスタンスへのアクセスを制限、フィルタ、監視、監査することが不可欠です。SUSEは、適切な境界セキュリティが欠如しているグローバルにアクセス可能なUyuniインスタンスを使用しないことを強くお勧めします。

UyuniのWeb UIにアクセスするには、ネットワークアクセス制御を設定するときにHTTPSを許可します。そうすると、UyuniのWeb UIにアクセスできます。

EC2およびAzureでは、新しいセキュリティグループを作成し、HTTPSの着信および受信のルールを追加します。GCEでは、[ファイアウォール]セクションで[HTTPSトラフィックを許可する]ボックスにチェックを付けます。

2.3.2. ストレージボリュームの準備

リポジトリとUyuniのデータベースは、ルートボリュームとは別のストレージデバイスに保存することをお勧めします。こうするとデータの損失が防止され、パフォーマンスが向上する可能性があります。

Uyuniコンテナはデフォルトのストレージの場所を利用します。これらの場所は、カスタムストレージ用の配備前に設定する必要があります。詳細については、[Installation-and-upgrade](#) > [Container-management](#)を参照してください



パブリッククラウドへのインストールでは論理ボリューム管理(LVM)を使用しないでください。

リポジトリストレージのディスクのサイズは、Uyuniで管理するディストリビューションおよびチャンネルの数によって決まります。仮想ディスクを接続すると、Unixデバイスノードとしてインスタンスに表示されます。デバイスノードの名前は、選択インスタンスの種類とプロバイダによって異なります。

Uyuniサーバのルートボリュームが100 GB以上であることを確認してください。500 GB以上のストレージディスクを追加し、可能な場合にはSSDストレージを選択します。Uyuniサーバのクラウドイメージは、スクリプトを使用して、インスタンス起動時にこの個別ボリュームを割り当てます。

インスタンスを起動すると、Uyuniサーバにログインし、次のコマンドを使用して、利用可能なすべてのストレージデバイスを検索できます。

```
hwinfo --disk | grep -E "デバイスファイル:"
```

選択したデバイスがわからない場合、**lsblk**コマンドを使用して、各デバイスの名前およびサイズを確認します。探している仮想ディスクのサイズと一致している名前を選択します。

mgr-storage-serverコマンドを使用して外部ディスクを設定できます。設定すると、XFSパーティションが/**manager_storage**にマウントされ、データベースおよびリポジトリの場所として使用されます。

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

章 3. 配備とインストール

3.1. Uyuniサーバのインストール

Uyuniサーバを配備するさまざまなシナリオがあります。

3.1.1. Uyuni Server Deployment on openSUSE Tumbleweed

3.1.1.1. 配備の準備

このセクションでは、Uyuniサーバのセットアップと配備に関する専門知識を身に付けることができます。このプロセスには、**Podman**と**Uyuniコンテナユーティリティ**のインストール、配備、および**mgrctl**を使用したコンテナとの対話の開始が含まれます。



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. コンテナホストの一般的な要件

一般的な要件については、**Installation-and-upgrade** › **General-requirements**を参照してください。

An openSUSE Tumbleweed server should be installed from installation media.

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

このプロシージャは以下で説明されます。

3.1.1.3. コンテナホストの要件

CPU、RAM、およびストレージの要件については、**Installation-and-upgrade** › **Hardware-requirements**を参照してください。



クライアントがFQDNドメイン名を解決できることを保証するには、コンテナ化されたプロキシとホストマシンの両方が、機能しているDNSサーバにリンクされている必要があります。さらに、リバース参照が正しく設定されていることを確認することも重要です。

3.1.1.4. コンテナで使用するためにUyuniツールをインストールする

Procedure: Installing Uyuni Tools on openSUSE Tumbleweed

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

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

3. Refresh the repository list and import the key:

```
zypper ref
```

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

4. コンテナツールをインストールします。

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

Uyuniコンテナユーティリティの詳細については、[Uyuniコンテナユーティリティ](#)を参照してください。

3.1.1.5. カスタム永続ストレージの設定

このステップはオプションです。ただし、ご使用のインフラストラクチャでカスタム永続ストレージが必要な場合は、**mgr-storage-server**ツールを使用します。

詳細については、**mgr-storage-server --help**を参照してください。このツールを使用すると、コンテナストレージとデータベースボリュームの作成が容易になります。

このコマンドは次のように使用します。

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

例:

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

 このコマンドは、**/var/lib/containers/storage/volumes**に永続ストレージを作成します。

詳細については、[Installation-and-upgrade](#) › [Container-management](#)を参照してください。

3.1.1.6. Podmanを使用したUyuniコンテナの配備

3.1.1.6.1. mgradmの概要

Uyuniは、**mgradm**ツールを使用してコンテナとして配備します。Uyuniサーバをコンテナとして配備する方法は2つあります。このセクションでは、基本的なコンテナ配備に焦点を当てます。

カスタム設定ファイルを使用した配備の詳細については、[Installation-and-upgrade](#) → [Container-management](#)を参照してください。

他の情報が必要な場合は、コマンドラインから`mgradm --help`を実行すると、詳しい情報を確認できます。

セキュリティ強化されたUyuniサーバホストでは、`/tmp`フォルダからのファイル実行が制限される場合があります。このような場合は、回避策として、`mgradm`を実行する前に、`TMPDIR`環境変数を別の既存のパスにエクスポートします。

 例:

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

Uyuniの更新では、この回避策が不要になるようにツールが変更されます。

プロシージャ: Podmanを使用してUyuniコンテナを配備する

1. 端末から、`sudo`ユーザまたはrootとして次のコマンドを実行します。

```
sudo mgradm install podman
```

コンテナは`sudo`またはrootとして配備する必要があります。このステップを省略すると、端末に次のエラーが表示されます。



```
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. 配備が完了するまで待ちます。
3. ブラウザを開き、サーバのFQDNに進みます。

3.1.1.6.2. 永続ボリューム

多くのユーザが永続ボリュームの場所を指定したいと考えています。

 Uyuniをテストしている場合は、これらのボリュームを指定する必要はありません。`mgradm`によって正しいボリュームがデフォルトでセットアップされます。

ボリュームの場所の指定は一般的に、大規模な運用配備で使用されます。

デフォルトでは、`podman`はそのボリュームを`/var/lib/containers/storage/volumes/`に保存します。

ディスクをこのパスにマウントするか、またはその内部の想定されるボリュームパス(`/var/lib/containers/storage/volumes/var-spacewalk`など)にマウントすることで、ボリュームにカスタムストレージを指定できます。これは特にデータベースとパッケージのミラーで重要です。

コンテナ内のすべての永続ボリュームのリストについては、以下を参照してください。

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

3.1.2. UyuniサーバのAir-gapped配備

3.1.2.1. Air-gapped配備とは

Air-gapped配備とは、安全ではないネットワーク、特にインターネットから物理的に隔離されたネットワークシステムをセットアップおよび運用することです。この種の配備は、一般的に高度なセキュリティ環境で使用されます。たとえば、軍事施設、金融システム、重要インフラストラクチャ、機密データを取り扱っていて外部の脅威から保護する必要がある場所などです。

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

Procedure: Pulling the images

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

リスト 1. Podman

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

リスト 2. Docker

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

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

リスト 3. サーバイメージをロードする

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

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

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

The images are available from **registry.opensuse.org**.

Procedure: Obtaining formula images for air-gapped environments

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

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

- Save the images to a TAR archive:

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

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

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

3.1.2.2.1. Deploy Uyuni on openSUSE Tumbleweed

Uyuniでは、システムにインストール可能なRPMパッケージで、必要なコンテナイメージもすべて提供します。



ユーザは必要なRPMを内部ネットワーク上で利用できるようにする必要があります。そのためには、2つ目のUyuniサーバまたは何らかの種類のミラーを使用します。

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

- Install openSUSE Tumbleweed.
- システムを更新します。
- ツールのパッケージとイメージパッケージをインストールします(\$ARCH\$は適切なアーキテクチャに置き換えます)。

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

- mgradm**を使用してUyuniを配備します。エアギャップ環境では、**--pullPolicy Never**オプションの使用をお勧めします。

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

Uyuniサーバをアップグレードするには、システム内のすべてのパッケージをアップグレードし、[サーバのアップグレード](#)で定義されている手順に従う必要があります。

3.2. Uyuniプロキシのインストール

Uyuniプロキシを配備するさまざまなシナリオがあります。これらのシナリオはすべて、すでにUyuni 2026.01サーバが正常に配備済みであることを前提としています。

3.2.1. コンテナ化されたUyuniプロキシの設定

Uyuniプロキシコンテナのコンテナホストが準備された後に、コンテナの設定では、設定を完了するための追加の手順がいくつか必要になります。

プロシージャ

1. Uyuniプロキシ設定アーカイブファイルを生成します
2. インストールステップで準備したコンテナホストに設定アーカイブを転送し、抽出します
3. **mgrpxy**でプロキシサービスを開始します

3.2.1.1. プロキシ設定の生成

Uyuniプロキシの設定アーカイブはUyuniサーバによって生成されます。追加のプロキシごとに専用の設定アーカイブが必要です。

コンテナ化されたUyuniプロキシの場合、変更を有効にするには、新しいプロキシ設定ファイルを構築してから、コンテナを再配備する必要があります。これは、SSL証明書を含む、設定を更新するためのプロセスです。



Podmanの配備では、このプロキシ設定を生成する前に、UyuniプロキシのコンテナホストをUyuniサーバにクライアントとして登録する必要があります。

プロキシFQDNを使用して、登録済みのクライアントではないプロキシコンテナ設定を生成する（Kubernetesのユースケースと同様）、新しいシステムエントリがシステム一覧に表示されます。この新しいエントリは、以前に入力されたプロキシFQDN値の下に表示され、**外部システムタイプ**になります。

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

On the hub server, run the following command.



```
mgrctl exec -ti -- rhn-ssl-tool --gen-server --dir="/root/ssl-build"
--set-country="COUNTRY" \
--set-state="STATE" --set-city="CITY" --set-org="ORGANIZATION" \
--set-org-unit="ORGANIZATION UNIT" --set-email="name@example.com" \
--set-hostname=PROXY --set-cname="proxy.example.com"
```

The files to use will be

1. **/root/ssl-build/RHN-ORG-TRUSTED-SSL-CERT** as the root CA,

- 2. `/root/ssl-build/<hostname>/server.crt` as the proxy certificate and
- 3. `/root/ssl-build/<hostname>/server.key` as the proxy certificate's key.

3.2.1.1.1. Web UIを使用したプロキシ設定の生成

プロシージャ: Web UIを使用してプロキシコンテナ設定を生成する

1. Web UIで、**システム** › **プロキシの設定**に移動し、必要なデータを入力します。
2. [プロキシFQDN] フィールドに、プロキシの完全修飾ドメイン名を入力します。
3. [親FQDN] フィールドに、Uyuniサーバまたは別のUyuniプロキシの完全修飾ドメイン名を入力します。
4. [プロキシSSHポート] フィールドに、SSHサービスがUyuniプロキシでリスンしているSSHポートを入力します。デフォルトの8022を維持することをお勧めします。
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.



2 GBはプロキシSquidのデフォルトのキャッシュサイズを表します。これは、環境に合わせて調整する必要があります。

[SSL証明書] 選択リストで、Uyuniプロキシ用に新しいサーバ証明書を生成するか、既存のサーバ証明書を使用するかを選択します。生成された証明書は、Uyuni組み込みの(自己署名)証明書とみなすことができます。

選択に応じて、新しい証明書を生成するための署名CA証明書へのパス、またはプロキシ証明書として使用される既存の証明書とそのキーへのパスのいずれかを指定します。

サーバによって生成されたCA証明書

は、`/var/lib/containers/storage/volumes/root/_data/ssl-build`ディレクトリに保存されます。

既存の証明書またはカスタム証明書、および企業証明書と中間証明書の概念の詳細については、Administration › Ssl-certs-importedを参照し

てください。

6. [生成]をクリックして、Uyuniサーバに新しいプロキシFQDNを登録し、コンテナホストの詳細を含む設定アーカイブ(**config.tar.gz**)を生成します。
7. しばらくすると、ダウンロードするファイルが表示されます。このファイルをローカルに保存します。

3.2.1.1.2. spacecmdと自己署名証明書を使用したプロキシ設定の生成

spacecmdを使用してプロキシ設定を生成できます。

プロシージャ: spacecmdと自己署名証明書を使用してプロキシ設定を生成する

1. SSHでコンテナホストに接続します。
2. 次のコマンドを実行してサーバとプロキシFQDNを置き換えます。

```
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. 生成された設定をサーバコンテナからコピーします。

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

3.2.1.1.3. spacecmdとカスタム証明書を使用したプロキシ設定の生成

デフォルトの自己署名証明書ではなくカスタム証明書に対して**spacecmd**を使用して、プロキシ設定を生成できます。

プロシージャ: spacecmdとカスタム証明書を使用してプロキシ設定を生成する

1. サーバコンテナホストにSSHで接続します。
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. 設定で中間CAを使用する場合は、それもコピーし、-iオプション付きでコマンドに含めます(必要に応じて複数回指定できます)。

```
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. 生成された設定をサーバコンテナからコピーします。

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

3.2.1.2. Uyuniプロキシ設定の転送

spacecmdコマンドとWeb UI経由の生成の両方の方法で、設定アーカイブが生成されます。このアーカイブは、コンテナホストで利用できるようにする必要があります。この生成されたアーカイブをコンテナホストに転送してください。

3.2.1.3. Uyuniプロキシコンテナの起動

mgrpxyコマンドを使用してコンテナを起動できます。

プロシージャ: Uyuniプロキシコンテナの起動

1. コマンドを実行します。

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

2. 次のコマンドを呼び出して、すべてのコンテナが期待どおりに起動したかどうかを確認します。

```
podman ps
```

5つのUyuniプロキシコンテナが存在し、**proxy-pod**コンテナポッドの一部である必要があります。

- proxy-salt-broker
- proxy-https
- proxy-tftpd
- proxy-squid
- proxy-ssh

3.2.2. Uyuni Proxy Deployment on openSUSE Tumbleweed

このガイドでは、Uyuni 2026.01プロキシの配備プロセスの概要について説明します。このガイドでは、Uyuni 2026.01サーバが正常に配備済みであることを想定しています。正常に配備するには、次のアクションを実行します。

チェックリスト: プロキシの配備

1. ハードウェア要件を確認します。
2. Install openSUSE Tumbleweed on a bare-metal machine.
3. プロキシをSalt Minionとしてブートストラップします。
4. プロキシ設定を生成します。
5. サーバからプロキシへのプロキシ設定の転送
6. プロキシ設定を使用して、Salt MinionをプロキシとしてUyuniに登録します。

プロキシコンテナホストでサポートされるオペレーティングシステム

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

コンテナホスト



コンテナホストは、コンテナを管理および配備できるPodmanなどのコンテナエンジンを搭載したサーバです。これらのコンテナは、アプリケーションと、ライブラリなどの重要な部品を保持しますが、完全なオペレーティングシステムは保持しないため軽量です。このセットアップにより、アプリケーションは異なる環境でも同じように動作します。CPU、メモリ、ストレージなど、これらのコンテナに必要なリソースはコンテナホストが提供します。

3.2.2.1. プロキシのハードウェア要件

次の表に、Uyuniプロキシを配備するためのハードウェア要件を示します。

表 9. プロキシのハードウェア要件

| ハードウェア | 詳細 | 推奨 |
|--------|-------------------------------------|--|
| CPU | x86-64、ARM | 専用64ビットCPUコア数は2つ以上 |
| RAM | 最小 | 2GB |
| | 推奨 | 8GB |
| ディスク容量 | / (ルートディレクトリ) | 40GB以上 |
| | /var/lib/containers/storage/volumes | 100GB以上。ストレージ要件は、使用するISOディストリビューションイメージ、コンテナ、およびブートストラップリポジトリの数に合わせて計算する必要があります。 |

3.2.2.2. コンテナホストの一般的な要件

一般的な要件については、[Installation-and-upgrade > General-requirements](#)を参照してください。

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

3.2.2.3. コンテナホストの要件

CPU、RAM、およびストレージの要件については、[Installation-and-upgrade › Hardware-requirements](#)を参照してください。



クライアントがFQDNドメイン名を解決できることを保証するには、コンテナ化されたプロキシとホストマシンの両方が、機能しているDNSサーバにリンクされている必要があります。さらに、リバース参照が正しく設定されていることを確認することも重要です。

3.2.2.4. コンテナで使用するためにUyuniツールをインストールする

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. ログインします。
3. Add the following repository to your openSUSE Tumbleweed server:

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

4. リポジトリのリストを更新してキーを受け入れます。

```
zypper ref
```

5. コンテナツールをインストールします。

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



または、**mgrpxy-zsh-completion**または**mgrpxy-fish-completion**をインストールできます。

Uyuniコンテナユーティリティの詳細については、[Uyuniコンテナユーティリティ](#)を参照してください。

3.2.2.5. カスタム永続ストレージの設定

このステップはオプションです。ただし、ご使用のインフラストラクチャにカスタム永続ストレージが必要な場合は、**mgr-storage-proxy**ツールを使用します。

詳細については、**mgr-storage-proxy --help**を参照してください。このツールを使用すると、コンテナストレージとSquidキャッシュボリュームの作成が容易になります。

このコマンドは次のように使用します。

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

例:

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

このコマンドは、**/var/lib/containers/storage/volumes**に永続ストレージを作成します。

 詳細については、以下を参照してください。

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

3.2.2.6. Minionとしてのプロキシホストのブートストラップ

タスク: プロキシホストのブートストラップ

1. システム › ブートストラップを選択します。
2. プロキシホストのフィールドに入力します。
3. ドロップダウンから、前のステップで作成したアクティベーションキーを選択します。
4. [+ ブートストラップ]をクリックします。
5. ブートストラッププロセスが正常に完了するまで待ちます。Saltメニューをチェックし、Salt Minionキーが一覧表示されていて受け入れられていることを確認します。
6. プロキシホストを再起動します。
7. すべてのイベントが終了したら、システムの一覧からホストを選択して2回目の再起動をトリガし、オンラインボーディングを完了します。

タスク: プロキシホストの更新

1. システムの一覧からホストを選択し、すべてのパッチを適用してホストを更新します。
2. プロキシホストを再起動します。

3.2.2.7. プロキシ設定の生成

Uyuniプロキシの設定アーカイブはUyuniサーバによって生成されます。追加のプロキシごとに専用の設定アーカイブが必要です。



このプロキシ設定を生成する前に、UyuniプロキシのコンテナホストをSalt Minionとして

てUyuniサーバに登録する必要があります。

次のタスクを実行します。

プロシージャ:

1. プロキシ設定ファイルを生成します。
2. 設定をプロキシに転送します。
3. **mgrpxy**コマンドでプロキシを起動します。

タスク: Web UIを使用したプロキシコンテナ設定の生成

1. Web UIで、**システム** › **プロキシの設定**に移動し、必要なデータを入力します。
2. **[プロキシFQDN]** フィールドに、プロキシの完全修飾ドメイン名を入力します。
3. **[親FQDN]** フィールドに、Uyuniサーバまたは別のUyuniプロキシの完全修飾ドメイン名を入力します。
4. **[プロキシSSHポート]** フィールドに、SSHサービスがUyuniプロキシでリスンしているSSHポートを入力します。デフォルトの8022を維持することをお勧めします。
5. **[Squidの最大キャッシュサイズ[MB]]** フィールドタイプで、Squidキャッシュの最大許容サイズ。通常、これはコンテナで使用可能なストレージの最大60%である必要があります。 **[SSL証明書]** 選択リストで、Uyuniプロキシ用に新しいサーバ証明書を生成するか、既存のサーバ証明書を使用するかを選択します。生成された証明書は、Uyuni組み込みの(自己署名)証明書とみなすことができます。

選択に応じて、新しい証明書を生成するための署名CA証明書へのパス、またはプロキシ証明書として使用される既存の証明書とそのキーへのパスのいずれかを指定します。

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

既存の証明書またはカスタム証明書、および企業証明書と中間証明書の概念の詳細については、**Administration** › **Ssl-certs-imported**を参照してください。

6. **[生成]**をクリックして、Uyuniサーバに新しいプロキシFQDNを登録し、コンテナホストの詳細を含む設定アーカイブを生成します。
7. しばらくすると、ダウンロードするファイルが表示されます。このファイルをローカルに保存します。

3.2.2.8. プロキシ設定の転送

Web UIで、設定アーカイブが生成されます。このアーカイブは、コンテナホストで利用できるようにする必要があります。

タスク: プロキシ設定のコピー

1. Copy the files to the Proxy host:

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

- 次のコマンドを使用してプロキシをインストールします。

```
mgrpwy install podman config.tar.gz
```

3.2.2.9. Uyuni 2026.01プロキシの起動

mgrpwyコマンドを使用してコンテナを起動できます。

タスク: プロキシの起動とステータスの確認

- 次のコマンドを呼び出してプロキシを起動します。

```
mgrpwy start
```

- 次のコマンドを呼び出してコンテナのステータスを確認します。

```
mgrpwy status
```

5つのUyuniプロキシコンテナが存在し、また、**proxy-pod**コンテナポッドの一部である必要があります。

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

3.2.2.9.1. サービスにカスタムコンテナイメージを使用する

デフォルトでは、Uyuniプロキシスイートは、その各サービスに対して同じイメージバージョンとレジストリパスを使用するように設定されています。ただし、末尾に**-tag**および**-image**を指定してインストールパラメータを使用し、特定のサービスのデフォルト値を上書きすることは可能です。

たとえば、次のように使用します。

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

これは、httpdサービスの設定ファイルを調整してから再起動します。**registry.opensuse.org/uyuni/proxy-
httpd**は使用するイメージ、**0.1.0**はバージョントグです。

値をデフォルトにリセットするには、これらのパラメータを指定せずにもう一度installコマンドを実行します。

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

このコマンドは、すべてのサービスの設定をグローバルデフォルトにリセットして再ロードします。

3.2.3. クライアントからのプロキシ変換

3.2.3.1. 概要

この章では、Web UIを使用してクライアントシステムをUyuniプロキシに変換する方法を説明します。

プロキシホストシステムは、すでにブートストラップが完了しており、ベースオペレーティングシステムチャネルにサブスクライブされていることを前提としています。

クライアントのオンボーディングの詳細については、[Client-configuration > Registration-overview](#)を参照してください。

3.2.3.2. 要件

変換を開始する前に、以下の要件が満たされていることを確認します。

3.2.3.2.1. クライアントは以下を満たしている必要があります

- Uyuniにすでにオンボーディング済みであること
- ネットワーク経由でアクセス可能であること

3.2.3.3. 準備

プロキシ変換を進める前に、変換プロセス中の中断を回避するため、以下の準備が完了していることを確認してください。

3.2.3.3.1. SSL証明書

プロキシと他のコンポーネント間の通信をセキュリティで保護するために、有効なSSL証明書が必要です。

以下のものが必要です。

- Uyuniサーバ上の証明書に署名した認証局(CA)の公開証明書
- プロキシの証明書。
- プロキシ証明書に対応する機密鍵。



CAが中間証明書チェーンを使用している場合は、すべて中間証明書も含める必要があります。

サードパーティ証明書を使用していない場合は、Uyuniコンテナ内の[rhn-ssl-tool](#)を使用して生成できます。

プロキシ証明書を生成する

- Uyuniサーバホストで、次のコマンドを実行します。

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

他のパラメータの詳細については、Administration > Ssl-certs-selfsignedを参照してください。

- Uyuniサーバホストに証明書を転送します。

```
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
```



証明書とキーファイルが生成された正確なフォルダを確認するには、次のコマンドでディレクトリを一覧表示できます。

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

- Uyuniサーバホストから証明書を転送します。

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

3.2.3.3.2. パッケージの準備

`mgrpx` のインストール

`mgrpx` ツールは、システムに一致するレポジトリからインストールする必要があります。以下から適切なレポジトリを選択します。

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

リスト 4. Example openSUSE Tumbleweed installation:

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

コンテナイメージのインストール

コンテナイメージをRPMパッケージとして配備することをお勧めします。次のパッケージがクライアントにインストールされていることを確認してください。

```
zypper ar
```

```
https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable/containerfile/
uyuni-proxy-images
zypper ref
zypper in uyuni-proxy-httpd-image \
    uyuni-proxy-salt-broker-image \
    uyuni-proxy-squid-image \
    uyuni-proxy-ssh-image \
    uyuni-proxy-tftpd-image
```

Air-gapped配備の詳細については、[Installation-and-upgrade](#) > [Container-deployment](#)を参照してください。

3.2.3.4. プロキシクライアントのセットアップ

1. クライアントの [\[概要\]](#) ページに移動します。
2. [\[プロキシへの変換\]](#)ボタンをクリックします。

プロキシ設定フォームにリダイレクトされたことを確認します。

このページには後で [\[詳細\]](#) > [\[プロキシ\]](#) > [\[設定\]](#) タブからアクセスできます。

3. Web UIで、[プロキシ](#)、[設定](#)に移動し、必要なデータを入力します。

プロシージャ: プロキシの設定

- a. [\[親FQDN\]](#) フィールドに、親サーバまたはプロキシの完全修飾ドメイン名を入力します。
- b. [\[プロキシSSHポート\]](#) フィールドに、SSHサービスがUyuniプロキシでリスンしているSSHポートを入力します。デフォルトの8022を維持することをお勧めします。
- c. [\[Squidの最大キャッシュサイズ\]](#) フィールドに、Squidキャッシュの最大許容サイズをギガバイト単位で入力します。
- d. [\[プロキシ管理者の電子メール\]](#) フィールドに、管理者の電子メールアドレスを入力します。
- e. [\[証明書\]](#) セクションに、準備ステップで取得したUyuniプロキシの証明書を指定します。
- f. [\[ソース\]](#) セクションで、2つのオプション([\[RPM\]](#) または [\[レジストリ\]](#))のいずれかを選択します。
 - [\[RPM\]](#) オプション、はエアギャップ環境または制限された環境に推奨されます。 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**.
 - [単純] を選択した場合は、 [\[レジストリURL\]](#) と [\[コンテナタグ\]](#) フィールドに値を入力します。
 - [レジストリURL] の場合は、 registry.opensuse.org/uyuniを使用します。
 - ドロップダウンリストからタグを選択します。
 - [詳細] を選択する場合は、フォームの追加セクションが表示されます。
 - 個々のコンテナURLフィールドごとに、レジストリを使用します。registry.opensuse.org/uyuniの後に対応するサフィックス(例: proxy-httpd` ま

たは`salt-broker)が続けます。

- ドロップダウンリストからタグを選択します。
4. すべてのフィールドが入力されたら、[適用]をクリックして設定を適用し、プロキシインストールタスクをスケジュールします。

3.2.3.5. プロキシの有効化の確認

クライアントのイベント履歴をチェックして、タスクの成功を確認します。

(オプション)プロキシのHTTPエンドポイントにアクセスして、ようこそページが表示されることを確認します。

3.2.4. K3sへのUyuniプロキシの配備

3.2.4.1. K3sのインストール

コンテナホストマシンに**K3s**をインストールします(<K3S_HOST_FQDN>はK3sホストのFQDNに置き換えます)。

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

3.2.4.2. ツールのインストール

インストールには**mgrpxy**パッケージと**helm**パッケージが必要です。

インストーラスクリプトを使用してHelmをインストールします。

```
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
```

詳細については、<https://helm.sh/docs/intro/install/#from-script>を参照してください。

mgrpxyパッケージはコンテナのutilsリポジトリで利用可能です。<https://download.opensuse.org/repositories/systemsmanagement:/Uyuni:/Stable:/ContainerUtils/>で、ディストリビューションに一致するパッケージを選択します。

プロシージャ

1. Leap Microにパッケージをインストールするには、次のコマンドを実行します。

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

2. 再起動します。

3.2.4.3. UyuniプロキシHelmチャートの配備

Uyuniプロキシポッドで使用するボリュームのストレージを設定するには、次のクレームに対して永続ボリュームを定義します。ストレージ設定をカスタマイズしない場合は、K3sによって自動的にストレージボリュームが作成されます。

永続ボリュームのクレームの名前は次のとおりです。

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

Installation-and-upgrade › **Container-deployment**に記載されているように、Uyuniプロキシの設定を作成します。設定**tar.gz**ファイルをコピーしてインストールします。

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

詳細については、以下を参照してください。

- <https://kubernetes.io/docs/concepts/storage/persistent-volumes/> (Kubernetes)
- <https://rancher.com/docs/k3s/latest/en/storage/> (K3s)ドキュメント

章 4. アップグレードと移行

4.1. サーバ

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. 要件と考慮事項

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



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

4.1.1.3. Migration Procedure

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

Procedure: Create a Backup

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

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

2. Package the backup for transfer:

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

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

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



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

4.1.1.3.2. Step 2: Reinstall the Host with openSUSE Tumbleweed

Procedure: Reinstalling the Host

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

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

Procedure: Installing Tools and Prerequisites

1. Add the Uyuni Stable repository and install tools:

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

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

```
zypper in podman
```



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

4.1.1.3.4. Step 4: Optional - Prepare Persistent Storage

Procedure: Preparing Persistent Storage

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

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

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



For details, see:

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

4.1.1.3.5. Step 5 Fetch and Restore the Backup on Tumbleweed

Procedure: Fetching and Restoring the Backup

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

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

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

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

4.1.1.3.6. Step 6: Start Services and Verify

Procedure: Starting Services and Verifying

1. Start the server services:

```
mgradm start
```

2. Verify:

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

4.1.1.4. Notes and Troubleshooting

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

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



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

4.1.2. レガシUyuniサーバからコンテナへの移行

レガシUyuniサーバをコンテナに移行するには、新しいマシンが必要です。

この移行のコンテキストでは、レガシUyuniサーバ(RPMインストール)は_古いサーバ_とも呼ばれます。

4.1.2.1. 要件と考慮事項

4.1.2.1.1. ホスト名

インプレース移行は不可能であるだけでなく、現在の移行プロシージャではホスト名の名前変更機能も許可されていません。

そのため、新しいサーバの完全修飾ドメイン名(FQDN)はレガシサーバのFQDNと同じままです。



移行後、新しいサーバを指すようにDHCPおよびDNSレコードを更新する必要があります。

詳細については、[移行を完了する](#)を参照してください。

4.1.2.1.2. SSL certificates

SSL証明書は後の段階で必要となります。自己署名生成CAおよび証明書を使用しない場合は、開始する前に以下があることを確認します。

- 認証局(CA) SSLパブリック証明書。CAチェーンを使用している場合は、すべての中間CAも利用できる必要があります。
- SSLデータベース機密鍵。
- SSLデータベース証明書。

すべてのファイルがPEM形式である必要があります。

SSLサーバ証明書のホスト名は、配備先マシンの完全修飾ホスト名と一致している必要があります。ホスト名は、証明書の**X509v3 Subject Alternative Name**セクションで設定できます。環境で必要な場合は、複数のホスト名を一覧にすることもできます。サポートされているキーの種類は、**RSA**と**EC** (Elliptic Curve)です。



データベースSSL証明書には **reportdb** と **db**、およびレポートデータベースにアクセスするするために使用される **Subject Alternative Name** としてのFQDNが必要です。

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キー

- 自己信頼GPGキーは移行されません。
- RPMデータベースで信頼されているGPGキーのみが移行されます。したがって、**spacewalk-repo-sync**でチャンネルを同期すると失敗する可能性があります。
- 管理者は、実際のサーバ移行後にこれらのキーをレガシUyuniインストールからコンテナホストに手動で移行する必要があります。

プロシージャ: GPGキーの新しいサーバへの手動移行

- レガシUyuniサーバから新しいサーバのコンテナホストにキーをコピーします。
- その後、コマンド**mgradm gpg add <PATH_TO_KEY_FILE>**を使用して、移行したサーバに各キーを追加します。

4.1.2.2.1. レガシサーバでの初期準備

移行には、複製が必要なデータの量によって、非常に長い時間がかかることがあります。ダウンタイムを短縮するには、レガシサーバ上のすべてのサービスを稼働させたまま、初期複製、再複製、_最終複製と切り替え_のプロセスで移行を複数回実行することができます。

最終的な移行中にのみ、レガシサーバ上のプロセスを停止する必要があります。

最終複製以外のすべてに対して、パラメータ--**prepare**を追加し、レガシサーバ上のサービスが自動的に停止するのを防止します。例:

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

プロシージャ: レガシサーバでの初期準備

- Uyuniサービスを停止します。

```
spacewalk-service stop
```

- PostgreSQLサービスを停止します。

```
systemctl stop postgresql
```

4.1.2.2.2. SSH接続の準備

プロシージャ: SSH接続を準備する

- 新しい2026.01サーバに**root**用のSSHキーが存在することを確認します。キーが存在しない場合は、次のコマンドで作成します。

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

- レガシサーバへの接続にパスワードを要求しないように、新しいサーバホストでSSH設定とエージェントの準備ができている必要があります。

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



パスワードを要求しない接続を確立するために、移行スクリプトは、新しいサーバで実行されているSSHエージェントに依存します。このエージェントがまだアクティブではない場合は、`eval $(ssh-agent)`を実行して開始します。その後、`ssh-add`に機密鍵のパスを指定して、実行中のエージェントにSSHキーを追加します。このプロセス中に、機密鍵のパスワードの入力を求められます。

- `ssh-copy-id`を使用して、SSH公開鍵をレガシUyuniサーバ(<oldserver.fqdn>)にコピーします。<oldserver.fqdn>は、レガシサーバのFQDNに置き換えてください。

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

SSHキーは、レガシサーバの`~/.ssh/authorized_keys`ファイルにコピーされます。 詳細については、`ssh-copy-id`のマニュアルページを参照してください。

- 新しいサーバからレガシUyuniサーバへのSSH接続を確立し、パスワードが不要であることを確認します。また、ホストの指紋に問題があつてはなりません。問題がある場合は、`~/.ssh/known_hosts`ファイルから古い指紋を削除してから、再試行してください。指紋はローカル`~/.ssh/known_hosts`ファイルに保存されます。

4.1.2.2.3. 移行の実行

レガシUyuniからコンテナ化されたUyuniへの移行を計画する際は、ターゲットのインスタンスがレガシ設定の仕様を満たしているか、上回っていることを確認します。これには、メモリ(RAM)、CPUコア、ストレージ、ネットワーク帯域幅などが含まれますが、これらに限定されません。

セキュリティ強化されたUyuniサーバホストでは、`/tmp`フォルダからのファイル実行が制限される場合があります。このような場合は、回避策として、`mgradm`を実行する前に、`TMPDIR`環境変数を別の既存のパスにエクスポートします。



例:

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

Uyuniの更新では、この回避策が不要になるようにツールが変更されます。

カスタム永続ストレージの設定

永続ストレージの設定はオプションですが、コンテナのディスクがいっぱいになった状態での深刻な問題を回避する唯一の方法です。`mgr-storage-server`ツールを使用してカスタム永続ストレージを設定することを

強くお勧めします。

詳細については、**mgr-storage-server --help**を参照してください。このツールを使用すると、コンテナストレージとデータベースボリュームの作成が容易になります。

このコマンドは次のように使用します。

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



デバイスにはファイルシステムが存在してはなりません。ファイルシステムがストレージデバイスに存在する場合、コマンドは中止されます。

例:

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

このコマンドは、*/var/lib/containers/storage/volumes*に永続ストレージを作成します。



詳細については、以下を参照してください。

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

移行の実行

1. 次のコマンドを実行して、新しいUyuniサーバをインストールします。**<oldserver.fqdn>**は、レガシサーバのFQDNに置き換えます。

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

2. 信頼されているSSL CA証明書を移行します。

証明書の移行

RPMの一部としてインストールされ、レガシUyuniの*/usr/share/pki/trust/anchors*ディレクトリに保存されている信頼済みSSL CA証明書は移行されません。SUSEはRPMパッケージをコンテナ内にインストールしないため、管理者はこれらの証明書ファイルを移行後にレガシインストールから手動で移行する必要があります。

プロシージャ: 証明書の移行

1. ファイルをレガシサーバから新しいサーバにコピーします。たとえば、**/local/ca.file**です。
2. 次のコマンドを使用してファイルをコンテナにコピーします。

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

移行を完了する

mgradm migrateコマンドを正常に実行した後で、すべてのクライアントのSaltセットアップが依然としてレガシサーバを指します。



新しい2026.01サーバにリダイレクトするには、新しいサーバの名前をインフラストラクチャレベル(DHCPおよびDNS)で変更して、レガシサーバと同じFQDNとIPアドレスを使用する必要があります。



移行中に問題が発生した場合は、古いシステムを再起動できます。rootとして、以下のコマンドを実行してPostgreSQLとspacewalkサービスを再起動してください。

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

4.1.2.3. Kubernetesの準備

特に移行ジョブはコンテナを最初から開始することを考慮して、**mgradm migrate**コマンドを使用して移行を実行する前に、**永続ボリューム**を事前定義する必要があります。

詳細については、**Installation-and-upgrade** > **Container-management**にあるこれらのボリュームの準備に関するインストールセクションを参照してください。

4.1.2.4. 移行

次のコマンドを実行して、新しいUyuniサーバをインストールします。<**oldserver.fqdn**>はレガシサーバの適切なFQDNに置き換えます。

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

または

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



mgradm migrateコマンドが正常に実行された後も、すべてのクライアントのSaltの設定はまだレガシサーバを指したままです。新しいサーバにリダイレクトするには、新しいサーバの名前をインフラストラクチャレベル(DHCPおよびDNS)で変更して、レガシサーバと同じFQDNとIPアドレスを使用する必要があります。

4.1.3. Uyuniサーバのアップグレード

アップグレードコマンドを実行する前に、ホストオペレーティングシステムを更新する必要があります。ホストオペレーティングシステムを更新すると、**mgradm**ツールなどのUyuniツールも更新されます。

プロシージャ: サーバのアップグレード

1. **zypper**を使用してソフトウェアリポジトリを更新します。

```
zypper ref
```

- transactional-updateを使用して利用可能な更新を適用します。

```
transactional-update
```

- 更新が適用されたら、rebootを実行して再起動します。
- Uyuniサーバコンテナは、次のコマンドを使用して更新できます。

```
mgradm upgrade podman
```

このコマンドは、コンテナのステータスを最新に更新してサーバを再起動します。

- 未使用のコンテナイメージをクリーンアップし、ディスク容量を解放します。

```
podman image prune -a
```

サードパーティSSL証明書を使用したアップグレード

サードパーティ証明書を使用している場合、データベースコンテナには、次のサブジエクト代替名(SAN)を持つSSL証明書が必要です。

- db
- reportdb
- 外部向けの完全修飾ドメイン名



メインコンテナとデータベースコンテナの両方に同じ証明書を使用できますが、これらのSANも必要です。

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.

特定のバージョンへのアップグレード



タグパラメータを指定しない場合、デフォルトで最新バージョンにアップグレードされます。特定のバージョンにアップグレードするには、タグパラメータと目的のイメージタグを指定します。

upgradeコマンドとそのパラメータの詳細については、次のコマンドを使用します。

```
mgradm upgrade podman -h
```

Air-gappedインストールでは、まずコンテナのRPMパッケージをアップグレードし、その後**mgradm**コマンドを実行します。

4.1.3.1. データベースバックアップボリューム

mgradm migrationまたは**mgradm upgrade**を使用したサーバ移行またはアップグレードにより、データベースバックアップを含むボリュームを作成できます。

PostgreSQLデータベースのバージョンを上げる際には、アップグレードを実行する前に古いデータベースを別の場所に保存する必要があります。この目的のために、**mgradm**はボリューム**var-pgsql-backup**を動的に作成します。移行またはアップグレードが完了し、ユーザが新しいシステムが期待通りに動作することを確認したら、このボリュームは安全に削除できます。

4.2. プロキシ

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. 要件と考慮事項

- 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

- Copy the Uyuni proxy configuration directory to a safe location:

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

- 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/'
```

- 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

- Reinstall the machine with openSUSE Tumbleweed (server profile recommended).
- 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

- From the Uyuni Web UI, obtain the system reactivation key for the existing proxy system record (Systems > Details > Reactivation).
- 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

- Add the Uyuni Stable repository and install tools:

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

- 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

- Copy back the saved configuration directory to the new host:

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

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

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

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

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

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



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

be required.

4.2.1.3.6. Step 6: Verify the Proxy

Procedure: Verifying the Proxy

- Check containers are running:

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

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

4.2.1.4. トラブルシューティング

- 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. レガシプロキシからコンテナへの移行

コンテナ化されたプロキシは、systemdのサービスセットによって管理されるようになりました。コンテナ化されたプロキシの管理には、**mgrpxy**ツールを使用してください。

このセクションは、**mgrpxy**ツールを使用してレガシ**systemd**プロキシを移行するのに役立ちます。

ホストOSがopenSUSE LeapからopenSUSE Leap Microに変更されたため、Uyuniの以前のリリースから2026.01へのインプレース移行は今後もサポートされません。

 従来の接続プロトコルはUyuni 2026.01以降ではサポートされなくなりました。以前のUyuniリリースから2026.01に移行する前に、従来のプロキシを含む既存の従来のクラウドアントをSaltに移行する必要があります。

4.2.2.1. Systemdを使用したレガシプロキシからコンテナ化されたプロキシへの移行

4.2.2.1.1. プロキシ設定の生成

プロシージャ: プロキシ設定の生成

- UyuniサーバのWeb UIにログインします。
- 左側のナビゲーションから、**システム › プロキシの設定**を選択します。
- プロキシのFQDNを入力します。元のプロキシホストと同じFQDNを使用します。

4. サーバのFQDNを入力します。
5. プロキシのポート番号を入力します。デフォルトのポート8022を使用することをお勧めします。
6. 証明書と機密鍵は、サーバコンテナホストの`/var/lib/containers/storage/volumes/root/_data/ssl-build/`にあります。
 - RHN-ORG-TRUSTED-SSL-CERT
 - RHN-ORG-PRIVATE-SSL-KEY
7. 次のコマンドを使用して証明書と鍵をマシンにコピーします。

```
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. [Choose File]を選択して、ローカルマシンを参照して証明書を選択します。
9. [Choose File]を選択して、ローカルマシンを参照して機密鍵を選択します。
10. CAのパスワードを入力します。
11. [生成]をクリックします。

4.2.2.1.2. 新しいホストへのプロキシ設定の転送

プロシージャ: プロキシ設定を転送する

1. サーバから、プロキシ設定が含まれる、生成されたtar.gzファイルを新しいプロキシホストに転送します。

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

2. 次のステップを実行する前に、レガシプロキシを無効にします。

```
spacewalk-proxy stop
```

3. 次のコマンドを使用して新しいプロキシを配備します。

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

4. 次のコマンドを使用して新しいプロキシを有効にします。

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

5. `podman ps`を実行して、すべてのコンテナが存在していて実行されていることを確認します。

```
proxy-salt-broker
proxy-httppd
proxy-tftpd
proxy-squid
```

proxy-ssh

4.2.2.2. UyuniプロキシをUyuni 2026.01コンテナ化プロキシに移行する

プロシージャ: Uyuniコンテナ化プロキシをUyuni 2026.01の新しいコンテナ化プロキシに移行する

1. 新しいマシンをブートし、openSUSE Leap Micro 6.1のインストールを開始します。
2. インストールを完了します。
3. システムを更新します。

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

4. **mgrpxy**と、オプションで**mgrpxy-bash-completion**をインストールします。

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

5. 再起動します。
6. **tar.gz**プロキシ設定をホストにコピーします。

4.2.2.3. Web UIを使用したパッケージのインストール

mgrpxyパッケージと**mgrpxy-bash-completion**パッケージは、Minionがブートストラップされてサーバに登録された後にWeb UIでインストールすることもできます。

プロシージャ: Web UIを使用したパッケージのインストール

1. インストール後、**管理** › **セットアップウィザード** → 製品ページから、SLE Micro 6.1親チャンネルとプロキシの子チャンネルが追加および同期されていることを確認します。
2. Web UIで、**システム** › **アクティベーションキー**に移動し、同期されたSLE Micro 6.1チャンネルにリンクされているアクティベーションキーを作成します。
3. **システム** › **ブートストラップ**ページを使用して、システムをMinionとしてブートストラップします。
4. 新しいマシンがオンボーディングされてシステムリストに表示されたら、システムを選択して、**システムの詳細** › **Install Package (パッケージのインストール)**ページに移動します。
5. パッケージ**mgrpxy**および**mgrpxy-bash-completion**をインストールします。
6. システムを再起動します。

4.2.2.4. spacecmdと自己署名証明書を使用したプロキシ設定の生成

spacecmdを使用してプロキシ設定を生成できます。

プロシージャ: [literal]` `spacecmd` と自己署名証明書を使用したプロキシ設定

の生成

1. SSHでコンテナホストに接続します。
2. 次のコマンドを実行してサーバとプロキシFQDNを置き換えます。

```
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. 生成された設定をプロキシにコピーします。

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

4. 次のコマンドを使用してプロキシを配備します。

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

4.2.2.5. spacecmdとカスタム証明書を使用したプロキシ設定の生成

デフォルトの自己署名証明書ではなくカスタム証明書に対して**spacecmd**を使用して、プロキシ設定を生成できます。



2 GBはプロキシSquidのデフォルトのキャッシュサイズを表します。これは、環境に合わせて調整する必要があります。

プロシージャ: spacecmdとカスタム証明書を使用したプロキシ設定の生成

1. サーバコンテナホストにSSHで接続します。
2. 次のコマンドを実行してサーバとプロキシFQDNを置き換えます。

```
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. 生成された設定をプロキシにコピーします。

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

4. 次のコマンドを使用してプロキシを配備します。

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

4.2.3. Uyuniプロキシのアップグレード

アップグレードコマンドを実行する前に、ホストオペレーティングシステムを更新する必要があります。ホストオペレーティングシステムを更新すると、**mgrpxy**ツールなどのUyuniツールも更新されます。

プロシージャ: プロキシのアップグレード

- zypper**を使用してソフトウェアリポジトリを更新します。

```
zypper ref
```

- transactional-update**を使用して利用可能な更新を適用します。

```
transactional-update
```

- 更新が適用されたら、**reboot**を実行して再起動します。

- podman**上で実行されるUyuniプロキシコンテナは、次のコマンドを使用して更新できます。

```
mgrpxy upgrade podman
```

または、Kubernetesクラスタ上で実行されるプロキシコンテナは、次のコマンドを使用して更新できます。

```
mgrpxy upgrade kubernetes
```

- podman**上で、未使用のコンテナイメージをクリーンアップしてディスク容量を解放します。

```
podman image prune -a
```

Kubernetesではイメージクリーンアップが自動的に処理されますが、Kubernetesディストリビューションによって異なります。

- i** 特定のバージョンにアップグレードする際にタグパラメータを指定しない場合、デフォルトで最新バージョンにアップグレードされます。特定のバージョンにアップグレードするには、タグパラメータと目的のイメージタグを指定します。
- !** 通常の状況下では、すべてのプロキシコンテナに同じタグを使用して一貫性を確保することを強くお勧めします。

Air-gappedインストールでは、まずコンテナのRPMパッケージをアップグレードし、その後**mgrpxy**

upgrade podmanコマンドを実行します。

4.3. クライアント

4.3.1. クライアントのアップグレード

クライアントは、基盤となるオペレーティングシステムのバージョン設定システムを使用します。 SUSEオペレーティングシステムを使用するクライアントの場合、UyuniのWeb UI内でアップグレードを実行できます。

クライアントのアップグレードの詳細については、[Client-configuration](#) › [Client-upgrades](#)を参照してください。

章 5. 基本的なサーバとプロキシの管理

5.1. Custom YAML Configuration and Deployment with mgradm

カスタムの **mgradm.yaml** ファイルを作成し、配備時に **mgradm** ツールでこのファイルを利用することができます。

コマンドラインパラメータまたは **mgradm.yaml** 設定ファイルで基本的な変数が指定されていない場合、**mgradm** によって入力を求められます。



セキュリティのため、コマンドラインパラメータを使用してパスワードを指定するのは避けることをお勧めします。代わりに、適切なパーミッションで設定ファイルを使用します。

プロシージャ: カスタム設定ファイルを使用して Podman で Uyuni コンテナを配備する

- 次の例のような **mgradm.yaml** という名前の設定ファイルを準備します。

```
# データベースのパスワード。デフォルトでランダムに生成されます
db:
  password: MySuperSecretDBPass

# CA証明書のパスワード
ssl:
  password: MySuperSecretSSLPASSWORD

# SUSEカスタマーセンターの資格情報
scc:
  user: ccUsername
  password: ccPassword

# 組織名
organization: YourOrganization

# 通知を送信する電子メールアドレス
emailFrom: notifications@example.com

# 管理者アカウントの詳細
admin:
  password: MySuperSecretAdminPass
  login: LoginName
  firstName: Admin
  lastName: Admin
  email: email@example.com
```

- 端末から root として次のコマンドを実行します。サーバの FQDN の入力はオプションです。

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



コンテナは sudo または root として配備する必要があります。このステップを省略す

ると、端末に次のエラーが表示されます。

```
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. 配備が完了するまで待ちます。
4. ブラウザを開き、FQDNまたはIPアドレスの入力に進みます。

5.2. コンテナの起動と停止

次のコマンドを使用して、Uyuni 2026.01サーバコンテナを再起動、起動、および停止できます。

Uyuni 2026.01サーバを再起動(**restart**)するには、次のコマンドを実行します。

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

サーバを起動(**start**)するには、次のコマンドを実行します。

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

サーバを停止(**stop**)するには、次のコマンドを実行します。

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

5.3. Uyuniで使用されるコンテナ

以下は、Uyuni 2026.01で使用されるコンテナのリストです。

表 10. サーバコンテナ

| コンテナ名 | 説明 |
|--------------------------|---------------------|
| uyuni-server | 主な製品コンテナ |
| uyuni-db | 製品のデータベースコンテナ |
| uyuni-hub-xmlrpc | ハブ配備用のXML-RPCゲートウェイ |
| uyuni-server-attestation | サーバCOCO認証 |

| コンテナ名 | 説明 |
|------------------------|-------------------------|
| uyuni-saline | Salt 可観測性のためのSalineコンテナ |
| uyuni-server-migration | 移行ヘルパーコンテナ |

表 11. プロキシコンテナ

| コンテナ名 | 説明 |
|-------------------------|------------------------------|
| uyuni-proxy-htpd | すべてのHTTP通信を処理する主なプロキシコンテナ |
| uyuni-proxy-squid | Squidキャッシュ |
| uyuni-proxy-salt-broker | Saltフォワーダ |
| uyuni-proxy-ssh | SSHフォワーダ |
| uyuni-proxy-tftpd | TFTPDからHTTPへのトランスレータおよびフォワーダ |

5.4. 永続コンテナボリューム

コンテナ内で行った変更は保持されません。永続ボリュームの外部で加えた変更は破棄されます。以下にUyuni 2026.01の永続ボリュームのリストを示します。

デフォルトのボリュームの場所をカスタマイズするには、**podman volume create**コマンドを使用して、ポッドの最初の起動前に、必要なボリュームを作成するようにします。



この表は、Helmチャートおよびsystemctlサービス定義の両方で示されているボリュームマッピングに正確に従っています。

5.4.1. サーバ

次のボリュームは、サーバ上のPodmanのデフォルトのストレージの場所に保存されます。

表 12. 永続ボリューム: Podmanのデフォルトストレージ

| ボリューム名 | ボリュームディレクトリ |
|----------------|--------------------------------------|
| Podman Storage | /var/lib/containers/storage/volumes/ |

表 13. 永続ボリューム: root

| ボリューム名 | ボリュームディレクトリ |
|--------|-------------|
| root | /root |

表 14. 永続ボリューム: var/

| ボリューム名 | ボリュームディレクトリ |
|------------------|-----------------------|
| var-cobbler | /var/lib/cobbler |
| var-salt | /var/lib/salt |
| var-pgsql | /var/lib/pgsql/data |
| var-pgsql-backup | /var/lib/pgsql-backup |
| var-cache | /var/cache |
| var-spacewalk | /var/spacewalk |
| var-log | /var/log |

表 15. 永続ボリューム: `srv/`

| ボリューム名 | ボリュームディレクトリ |
|---------------------|-----------------------|
| 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 |

表 16. 永続ボリューム: `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 |

| Volume Name | Volume Directory |
|-------------|------------------------|
| ca-cert | /etc/pki/trust/anchors |

表 17. 永続ボリューム: run/

| ボリューム名 | ボリュームディレクトリ |
|-----------------|------------------|
| run-salt-master | /run/salt/master |

5.4.2. プロキシ

次のボリュームは、プロキシ上の**Podman**のデフォルトのストレージの場所に保存されます。

表 18. 永続ボリューム: Podmanのデフォルトストレージ

| ボリューム名 | ボリュームディレクトリ |
|----------------|--------------------------------------|
| Podman Storage | /var/lib/containers/storage/volumes/ |

表 19. 永続ボリューム: srv/

| ボリューム名 | ボリュームディレクトリ |
|----------------------|---------------|
| uyuni-proxy-tftpboot | /srv/tftpboot |

表 20. 永続ボリューム: var/

| ボリューム名 | ボリュームディレクトリ |
|-------------------------|------------------|
| uyuni-proxy-rhn-cache | /var/cache/rhn |
| uyuni-proxy-squid-cache | /var/cache/squid |

5.5. **mgr-storage-server**および**mgr-storage-proxy**について

mgr-storage-serverおよび**mgr-storage-proxy**は、Uyuniで提供されるヘルパースクリプトです。

これらはUyuniサーバおよびプロキシのストレージを設定するように設計されています。

スクリプトはディスクデバイスを引数として受け取ります。**mgr-storage-proxy**にはストレージディスクデバイス用の引数が1つ必要です。**mgr-storage-server**にはストレージディスクデバイスが必要であり、オプションで専用データベースディスクデバイス用の2番目の引数を受け取ることができます。通常のストレージとデータベースストレージは同一ディスク上に配置可能ですが、パフォーマンスの向上と管理の容易さを確保するため、データベースは専用の高性能ディスクに配置することをお勧めします。

5.5.1. これらのツールの機能

mgr-storage-serverと**mgr-storage-proxy**の両方が標準のストレージセットアップ操作を実行します。

- 提供されたストレージデバイスを検証します。
- デバイスが空で使用に適していることを確認します。
- 指定されたデバイス上にXFSファイルシステムを作成します。
- データ移行のためデバイスを一時的にマウントします。
- 関連するストレージディレクトリを新しいデバイスに移動します。
- ブート時にストレージが自動的にマウントされるように、**/etc/fstab**にエントリを作成します。
- デバイスを最終的な場所で再マウントします。

表 21. 追加のツール固有の動作

| | |
|---------------------------|---|
| mgr-storage-server | <ul style="list-style-type: none"> • オプションでデータベースストレージ用の個別のデバイスをサポートします。 • 移行中にSUSE Managerサービスを停止し、その後再起動します。Podmanボリュームディレクトリ/var/lib/containers/storage/volumesを準備済みストレージに移動し、オプションで/var/lib/containers/storage/volumes/var-pgsqlを準備済みデータベースストレージに移動します。 |
| mgr-storage-proxy | <ul style="list-style-type: none"> • プロキシストレージのみに焦点を当てます(データベースストレージはサポートしていません)。 • 移行中にプロキシサービスを停止し、再起動します。 • podmanボリュームディレクトリ/var/lib/containers/storage/volumesを準備済みストレージに移動します。 |



どちらのツールも標準的なLinuxストレージ操作を自動化します。Linx管理者が手動で行う範囲を超えた、隠れたロジックやカスタムロジックはありません。

5.5.2. これらのツールが実行しないこと

- LVMボリュームを作成または管理しません。
- RAIDや複雑なストレージトポロジを設定しません。
- セットアップ後に通常のLinuxツールを使用したストレージの管理を妨げることはありません。
- 動的なサイズ変更や拡張機能は提供しません。これらは標準的なLinuxストレージツールで処理する必要があります。

5.5.3. インストール後のストレージ管理

ストレージが設定されると、標準的なLinuxコマンドを使用して安全に管理できます。

5.5.3.1. 例

リスト 5. 例1: LVMを使用する場合のストレージの拡張

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

例2: 大容量ディスクへの移行

1. 新規ディスクを追加してフォーマットします。
2. 一時的にマウントします。
3. **rsync**を使用してデータをコピーします。
4. **/etc/fstab**を更新します。
5. 正しい場所に再マウントします。

5.5.4. 使用する場合、使用しない場合



ストレージセットアップに変更を行う前に、必ずバックアップを取ってください。

- これらのツールは初期ストレージセットアップ中、またはツールがデータ移行と**/etc/fstab**の更新を処理することが想定される新しいストレージへの移行時に**のみ**使用します。
- ストレージのサイズ変更や拡張のためにこれらのスクリプトを再実行しないでください。このような操作には標準的なLinuxツール(**lvextend**、**xfs_growfs**など)を使用してください。

5.5.5. 概要

mgr-storage-serverおよび**mgr-storage-proxy**は、標準的なLinuxストレージ手法を使用して、Uyuniコンポーネントの初期永続化ストレージセットアップを自動化するのに役立ちます。これらは、その後の標準的なストレージ管理を制限したり妨げたりすることはありません。

セットアップ後は、使い慣れたLinuxツールを使用してストレージの管理を続行してください。



データベースボリュームがいっぱいになると、システム動作に重大な問題を引き起こす可能性があります。ディスク使用量の通知はまだコンテナ化された環境には適応されていないため、ユーザはPodmanボリューム自体で使用されるディスク容量を、GrafanaやPrometheusなどのツールまたは他の好ましい方法で監視することをお勧めします。**/var/lib/containers/storage/volumes/**の下にあるvar-pgsqlボリュームに特に注意してください。

章 6. GNU Free Documentation License

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

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum

below.

- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this

License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".