Open Source MANO by ETSI

OSM Release FOURTEEN Webinar

Francisco-Javier Ramón (Telefónica, ETSI OSM Chair)
Gerardo García (Telefónica, TSC Chair)
Sergio Tarazona (WhiteStack, TSC Member)
Selvi Jayaraman (Tata Elxsi)

12/09/2023

Some requirements for the evolution of Telco Clouds…

Open Source MANO

NEXT-GEN TELCO CLOUD

REQUIREMENTS FROM NEW SERVICES

BUSINESS DRIVERS

Cost reduction

Time to market reduction

Improved customer experience

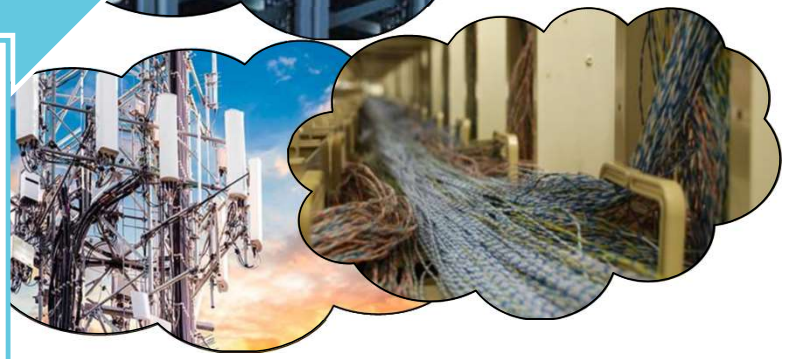
CONSTRAINTS

Network & IT, in process of convergence

Legacy systems (e.g., preexisting OSS)

Legacy Vertical Deployments

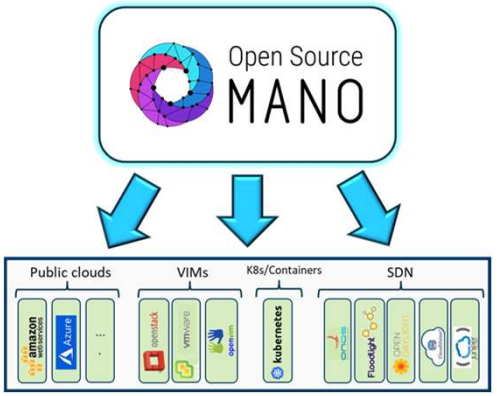
NEW TECHNOLOGY ENABLERS



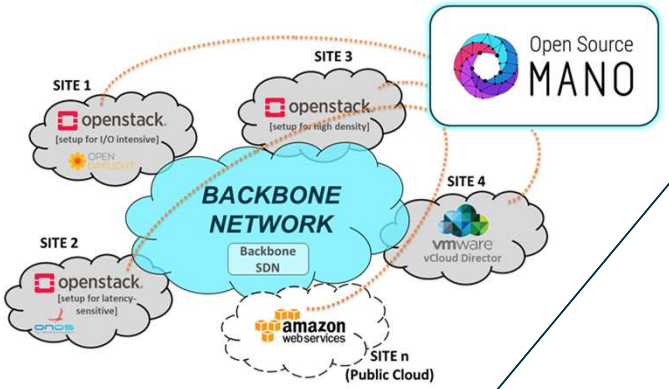
… on different types of infrastructure and across different locations…

Open Source MANO

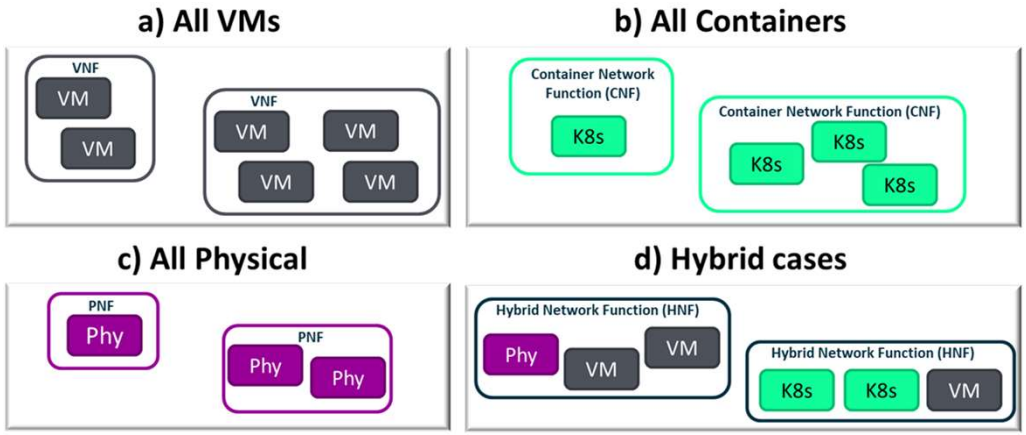
MULTI-VIM & MULTI-SDN



MULTI-SITE



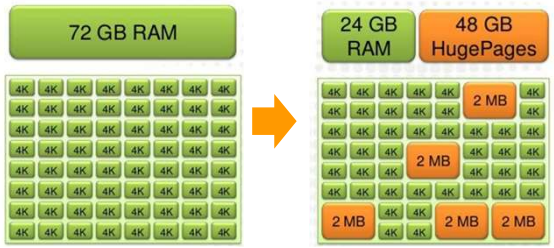
… with NFs composed of **VMs, containers** and/or **physical elements**…



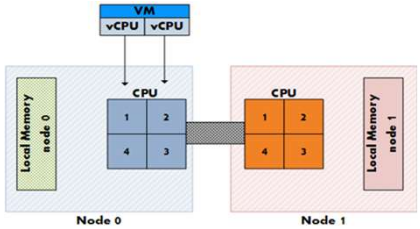
… and ready for network-specific workloads whenever needed

Open Source MANO

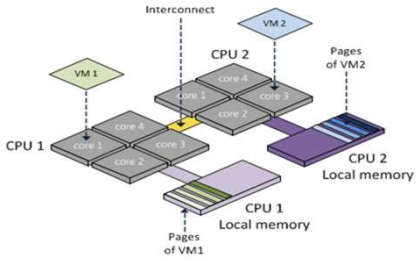
Huge Pages



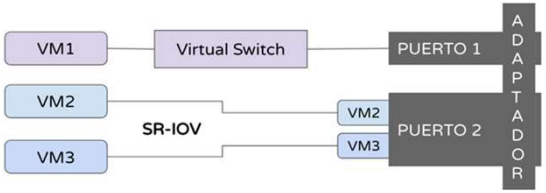
NUMA Topology Awareness

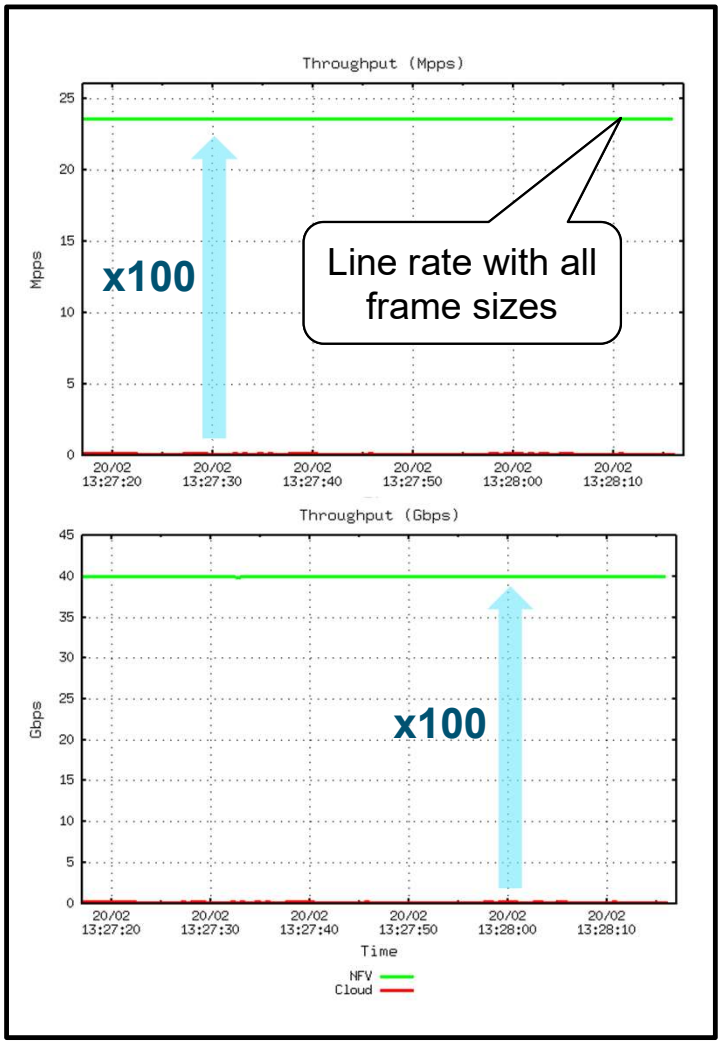


CPU Pinning

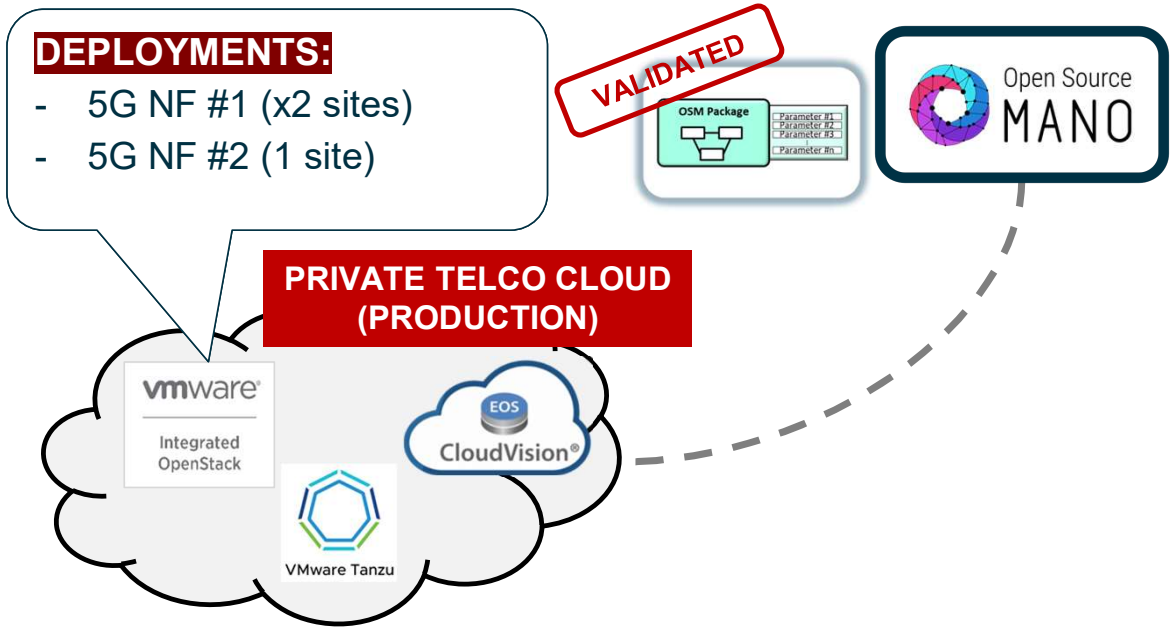


Data Plane assignment

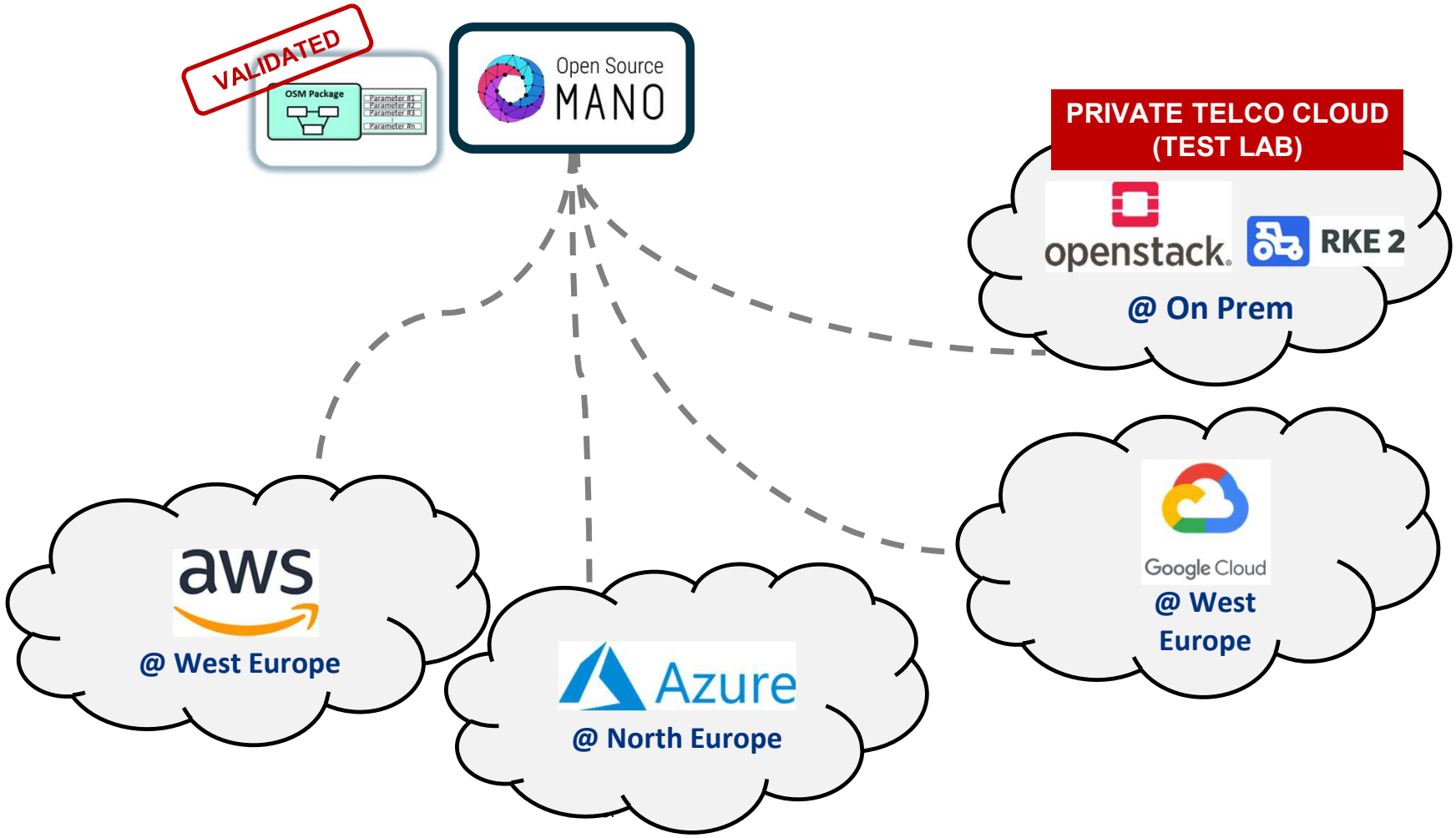




Using the exact same packages, the same service can be deployed in multiple types of clouds and sites



Using the exact same packages, the same service can be deployed in multiple types of clouds and sites



As a result, OSM brings big operational benefits for the challenges of a modern Telco Cloud

Open Source MANO

Reduction of complexity
- Via abstraction & layering

Reliable deployment in multiple locations

Independent of the type of cloud

Vendor-agnostic

Reliable and unambiguous testing
- Ideal for CI/CD

Error minimization

Minimal **Time to Market** for second deployments

Easier **capacity growth among clouds**

Ability to **move workloads** between clouds

Allows for advanced **redundancy schemas**

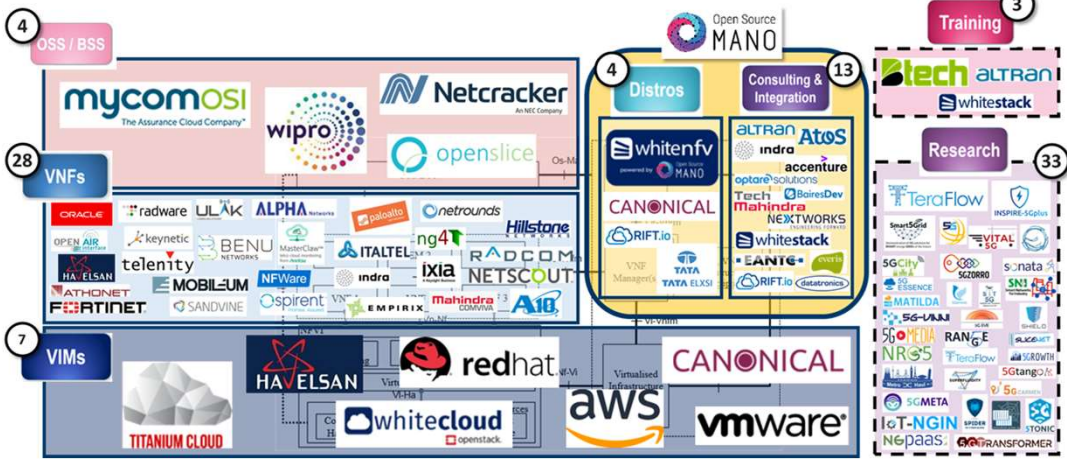
Reduction of efforts

Open Source MANO

A Vibrant and Thriving Community

ETSI OSM community is really LARGE AND DIVERSE, with 155 members today

… with a significant number of commercial offers related to OSM ("OSM Ecosystem")



https://osm.etsi.org/wikipub/index.php/OSM_Ecosystem

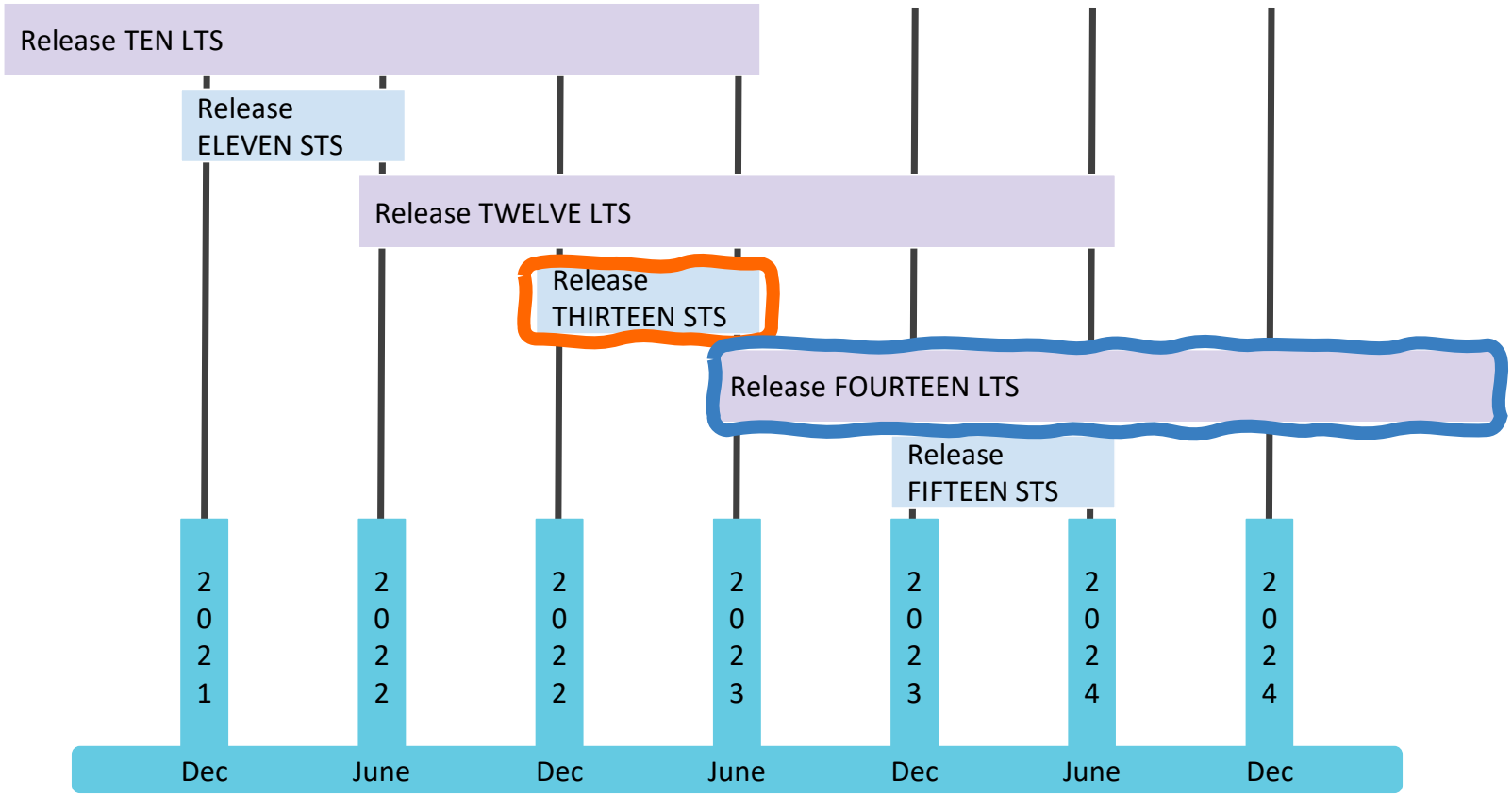
Open Source MANO

And the new release!

Open Source MANO

Release FOURTEEN

Rel FOURTEEN will be available as LTS



LTS Releases (Long Term Support)	STS Releases (Short Term Support)
24 months community support	6 months community support
Oriented to production	Oriented to innovation & development
Focus on stability	Focus on innovation & agility
Community grants upgrade between LTS's	Upgrade on a best effort basis

Release FOURTEEN brings a whole set of new functionalities

Closed-loop life cycle architecture

- GA of new monitoring architecture for closed loops.
- Service KPI of VNF using exporter endpoint.
- Autoheal switch and autoscale switch.

Security enhancements

- Replacement Pycrypto with PycryptoDome.
- Pod Admission Policy for Helm-based EE.
- Authenticated gRPC for Helm-based EE.

Usability and platform management

- User management enhancements.
- Audit logs generation for OSM

Infra modelling and NF lifecycle

- RO performance optimization.
- Simultaneous IPv4 and IPv6 support.
- Transport API (TAPI) WIM connector
- Support of volume multi-attach.
- Use existing flavor-id as an instantiation parameter.
- Instantiation parameters for Juju bundles.

OSM installation

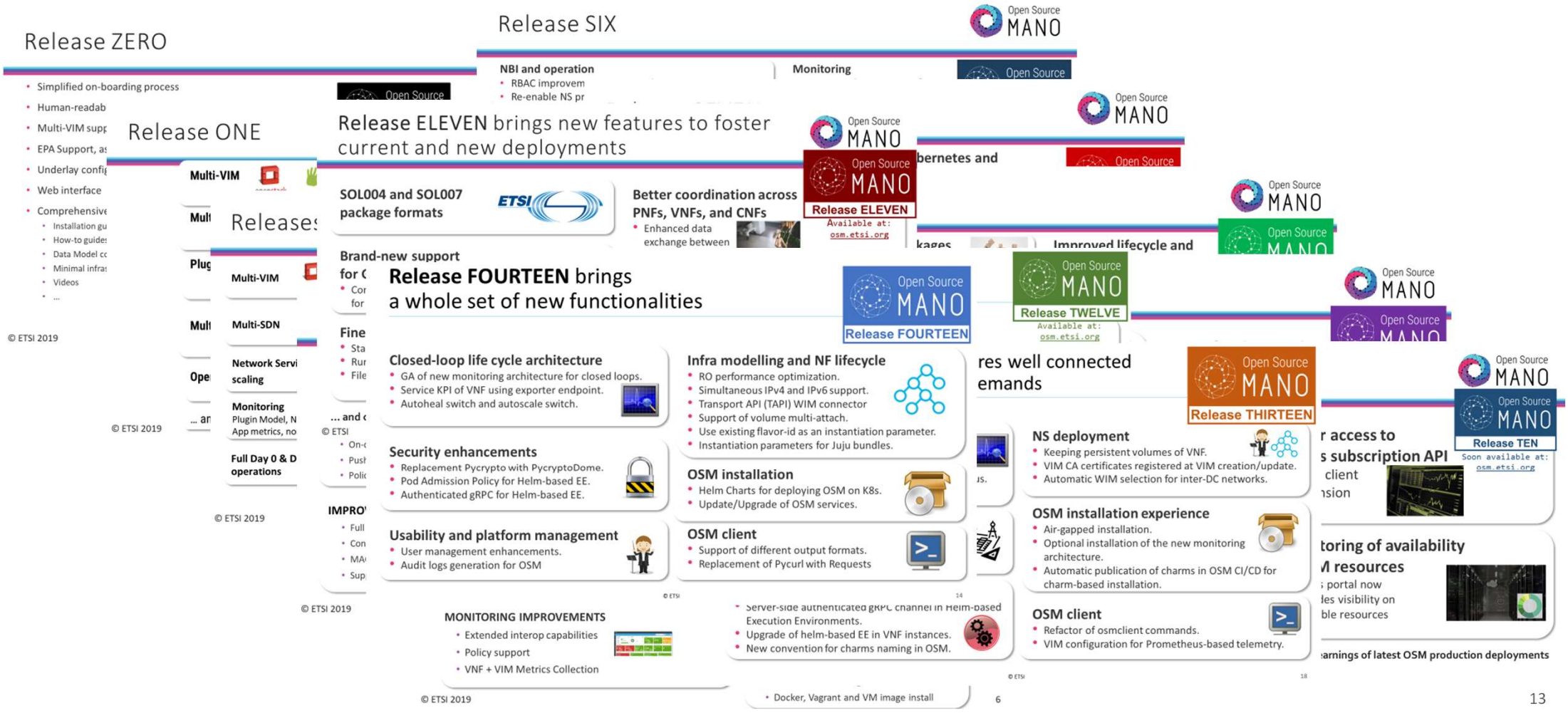
- Helm Charts for deploying OSM on K8s.
- Update/Upgrade of OSM services.

OSM client

- Support of different output formats.
- Replacement of Pycurl with Requests

These features are added on top of an already long set of features...

Open Source MANO



Agenda

- GA of new closed-loop architecture.
- Secured Helm-based execution environments.
- Improved user management.

Open Source MANO

GA New Closed-Loop Architecture

Overview and demo

Release FOURTEEN comes with the new closed-loop architecture enabled by default

- MON and POL functionality has been transferred to the new architecture
  - Metric acquisition (VM resource consumption)
  - Closed-loop for auto-healing
  - Closed-loop for auto-scaling
  - VNF alarms
- Airflow, Prometheus AlertManager and PushGateway are deployed
  - Airflow provide a mechanism to run scheduled workflows, as well as workflows on demand from an alert
- Webhook Translator is a new component that has been added to translate webhooks

The new SA architecture is installed by default

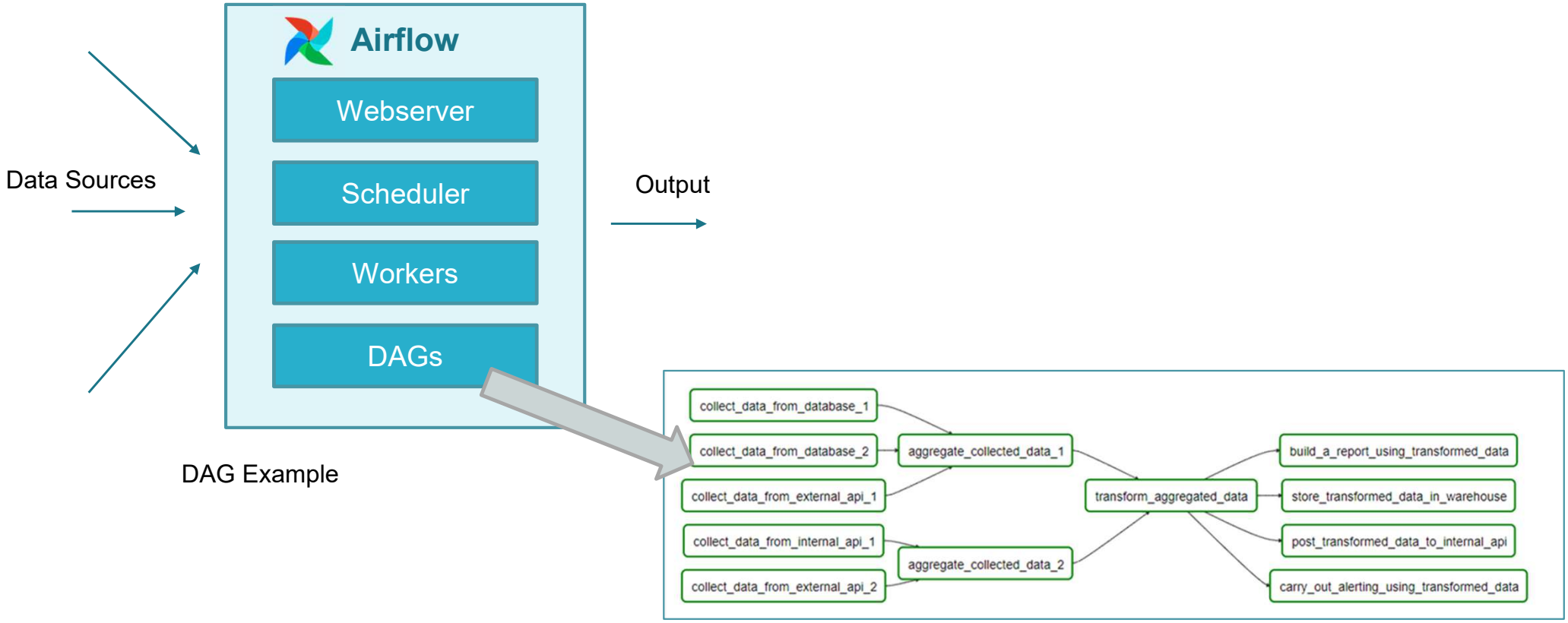
./install_osm.sh

```
$ helm -n osm ls
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART                           APP VERSION
airflow         osm             1               2023-06-07 15:08:48.613039036 +0000 UTC deployed        airflow-1.9.0                   2.5.3
alertmanager    osm             1               2023-06-07 15:10:23.448079581 +0000 UTC deployed        alertmanager-0.22.0             v0.24.0
osm             osm             1               2023-06-07 15:08:43.421836769 +0000 UTC deployed        osm-0.0.1                       14
pushgateway     osm             1               2023-06-07 15:10:19.507304535 +0000 UTC deployed        prometheus-pushgateway-1.18.2   1.4.2
```

```
$ kubectl -n osm get pods
NAME                                            READY   STATUS    RESTARTS        AGE
airflow-postgresql-0                            1/1     Running   2 (2d20h ago)   5d22h
airflow-redis-0                                 1/1     Running   1 (2d20h ago)   5d22h
airflow-scheduler-5f7dbdc4f5-54x9c              2/2     Running   4 (2d20h ago)   5d22h
airflow-statsd-d8c8f886c-vt7xq                  1/1     Running   4 (2d20h ago)   5d22h
airflow-triggerer-6668bd965c-n6snh              2/2     Running   3 (2d20h ago)   5d22h
airflow-webserver-5fb957dcf7-bcgzw              1/1     Running   1 (2d20h ago)   5d22h
airflow-worker-0                                2/2     Running   2 (2d20h ago)   5d22h
alertmanager-0                                  1/1     Running   6 (2d20h ago)   5d22h
grafana-69c9c55dfb-jtwfl                        2/2     Running   2 (2d20h ago)   5d22h
kafka-0                                         1/1     Running   1 (2d20h ago)   5d22h
keystone-7dbf4b7796-rqwg4                       1/1     Running   1 (2d20h ago)   5d22h
lcm-6d97b88675-4m77j                            1/1     Running   2 (2d20h ago)   5d22h
modeloperator-7dd8bf6c79-wx49m                  1/1     Running   1 (2d20h ago)   5d22h
mon-ccb965d54-drvmr                             1/1     Running   1 (2d20h ago)   5d22h
mongodb-k8s-0                                   1/1     Running   3 (2d20h ago)   5d22h
mongodb-k8s-operator-0                          1/1     Running   1 (2d20h ago)   3d11h
mysql-0                                         1/1     Running   1 (2d20h ago)   5d22h
nbi-64b4f6ffd9-jtbf5                            1/1     Running   5 (2d20h ago)   5d22h
ngui-78d9bd66dc-xbff6                           1/1     Running   3 (2d19h ago)   5d22h
prometheus-0                                    2/2     Running   4 (2d20h ago)   5d22h
pushgateway-prometheus-pushgateway-6f9dc6cb4d-4sp4x   1/1     Running   1 (2d20h ago)   5d22h
ro-86cf9d4b55-z6ls7                             1/1     Running   5 (2d20h ago)   5d22h
webhook-translator-57b75fc797-j9s7w             1/1     Running   1 (2d20h ago)   5d22h
zookeeper-0                                     1/1     Running   1 (2d20h ago)   5d22h
```

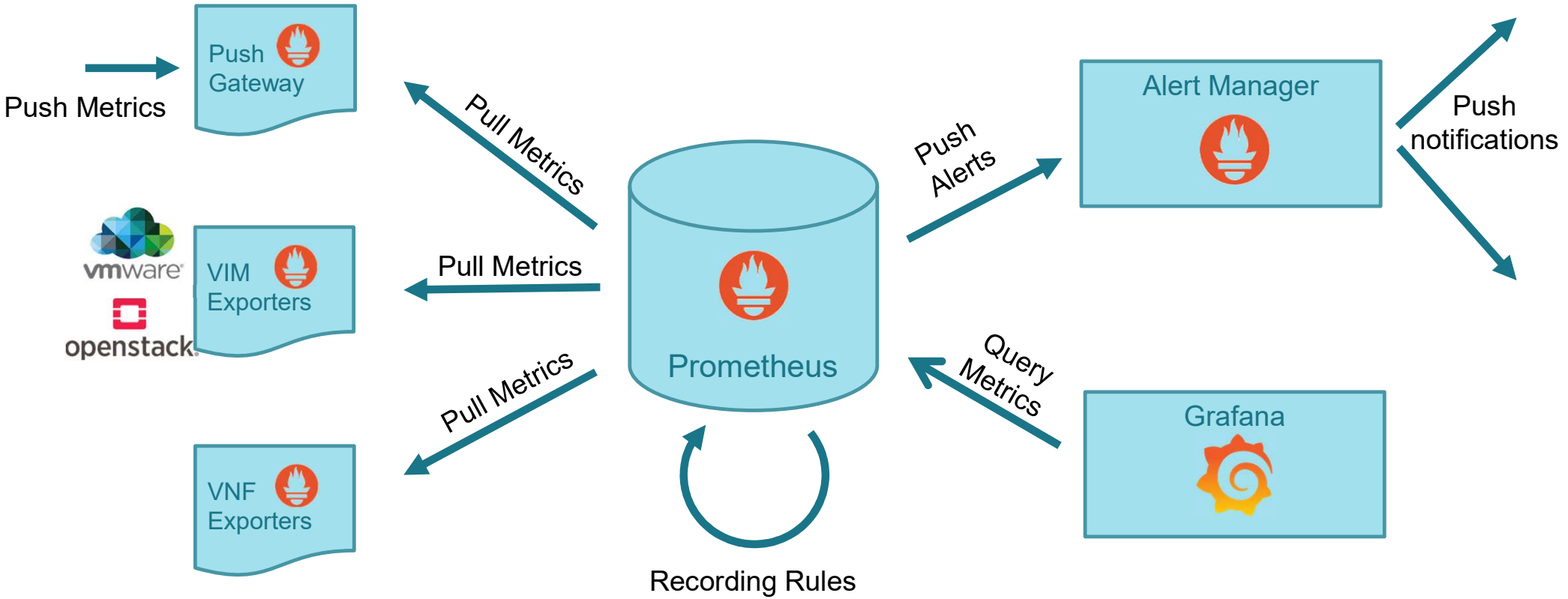
Building blocks of the new SA architecture

Building blocks of the new SA architecture
Apache Airflow

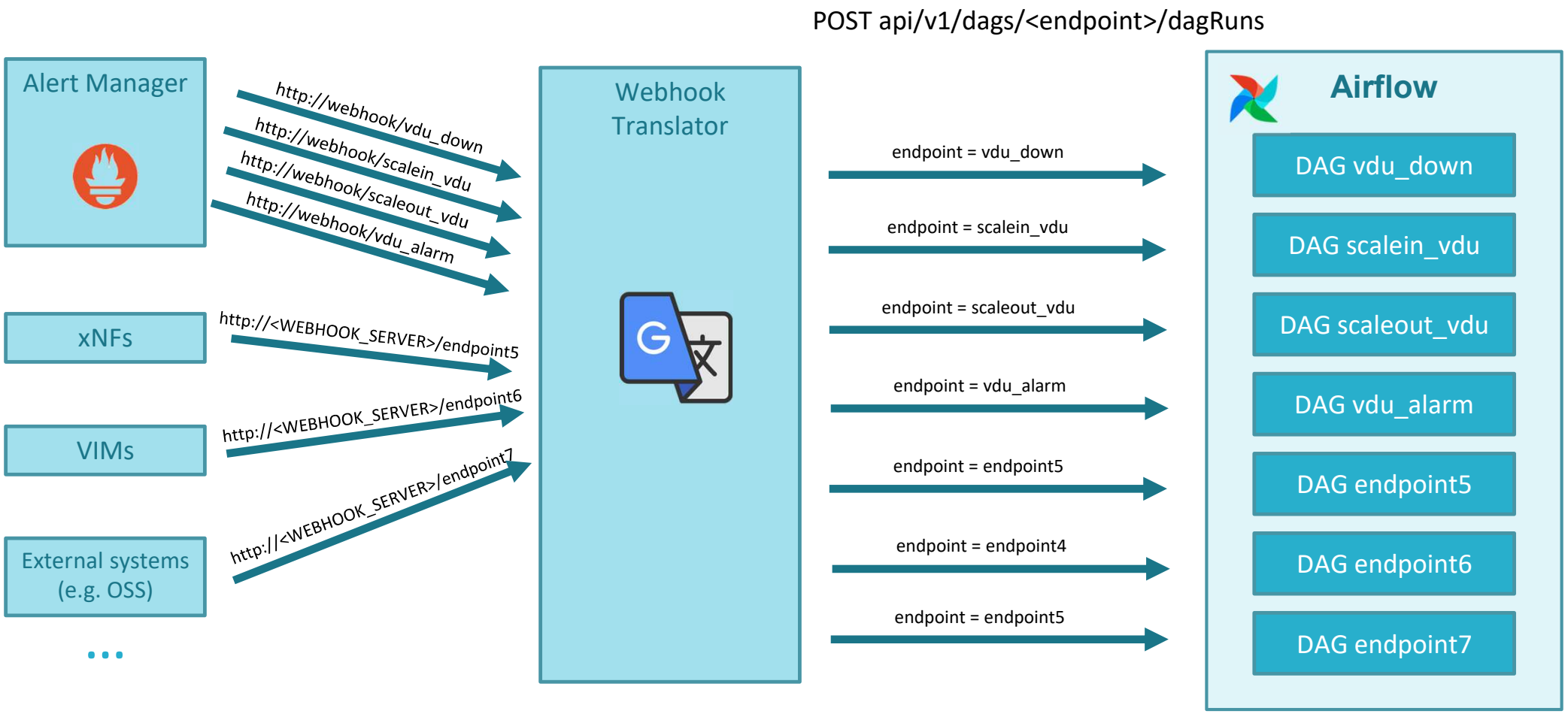


DAG Example

Building blocks of the new SA architecture
Prometheus Stack



Building blocks
Webhook translator



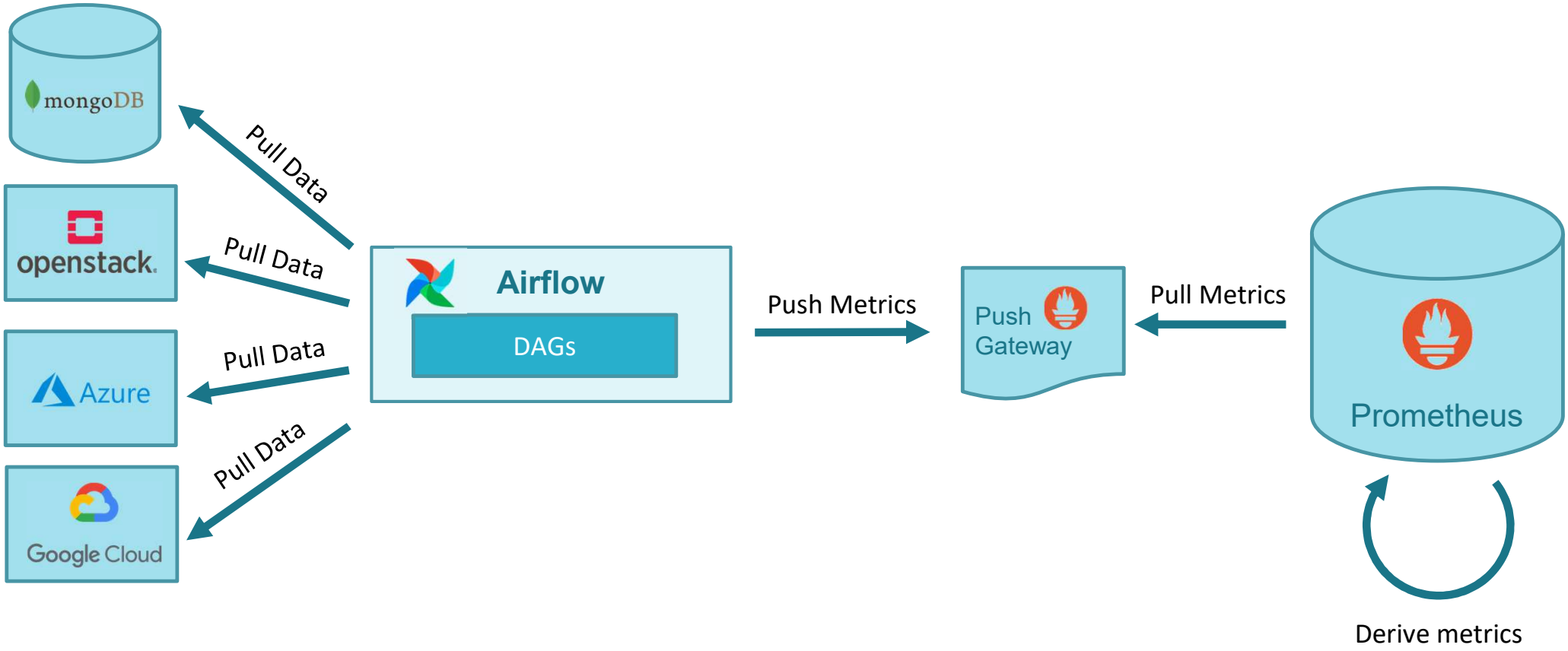
Building blocks
Webhook translator

- Principles:
  - Lightweight: a very small number of lines of code will do the work.
  - Stateless. It only translates HTTP requests. No state for those translations
    - When running as a deployment, native scaling is achieved by means of Kubernetes services
  - Simple. Based on FastAPI (https://fastapi.tiangolo.com/)
    - Simple and fast framework for developing an HTTP REST API in Python.
  - Independent from the source of the alert
    - No maintenance

Open Source MANO

Demo. Workflows

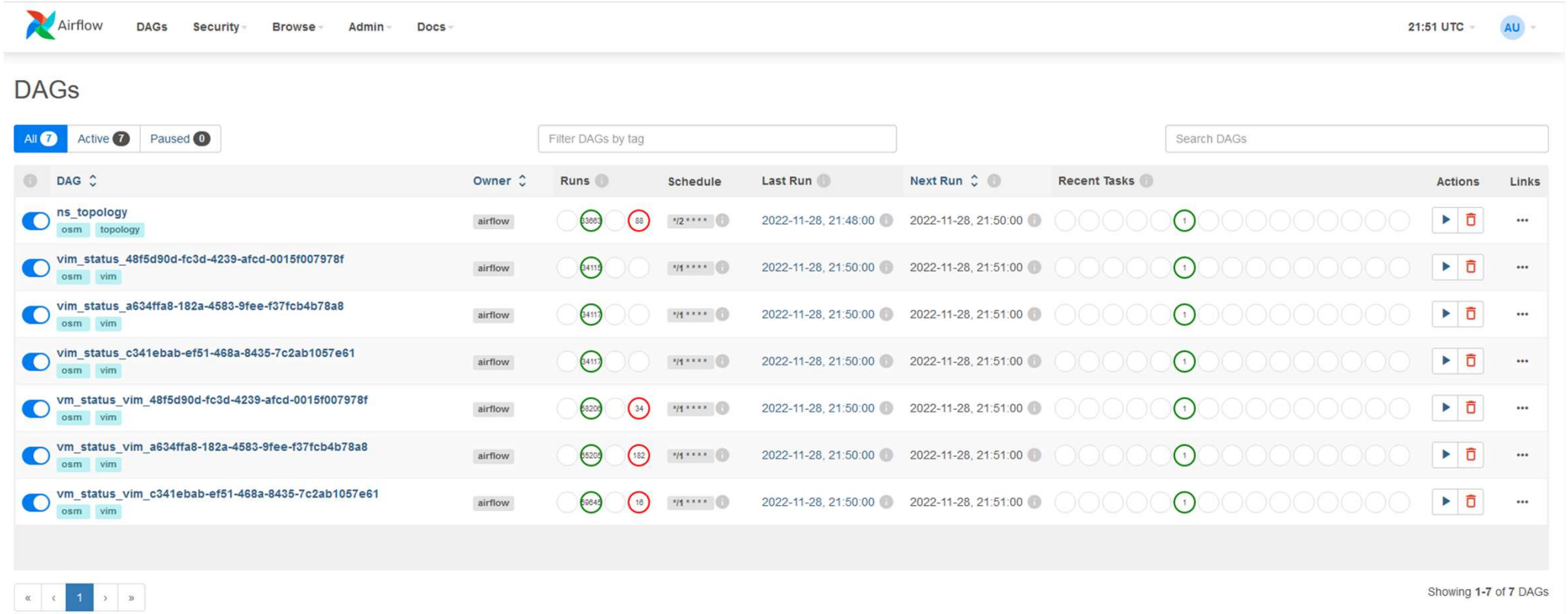
Workflow for metric acquisition and derivation



Metric acquisition

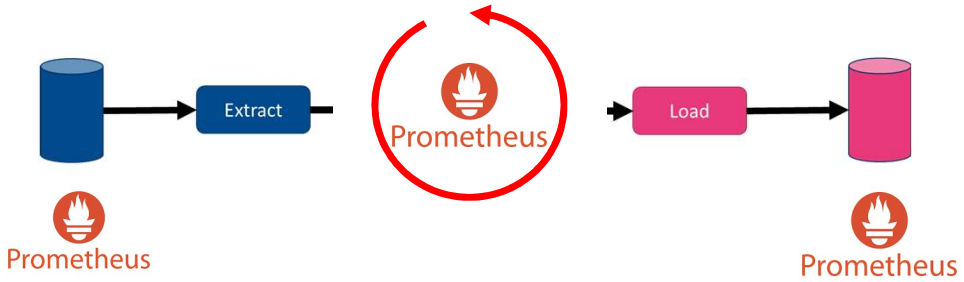
- NS topology:
  - From Mongo DB to Prometheus
  - SW used: Airflow DAG + Prometheus PushGateway
- VM status:
  - From MongoDB and VIM to Prometheus
  - SW used: Airflow DAG per VIM + Prometheus PushGateway
- VIM status
  - From MongoDB and VIM to Prometheus
  - SW used: Airflow DAG per VIM + Prometheus PushGateway
- VM metrics (resource consumption)
  - From MongoDB and VIM to Prometheus
  - SW used: Airflow DAG per VIM + Prometheus PushGateway

Screenshot of Airflow DAGs

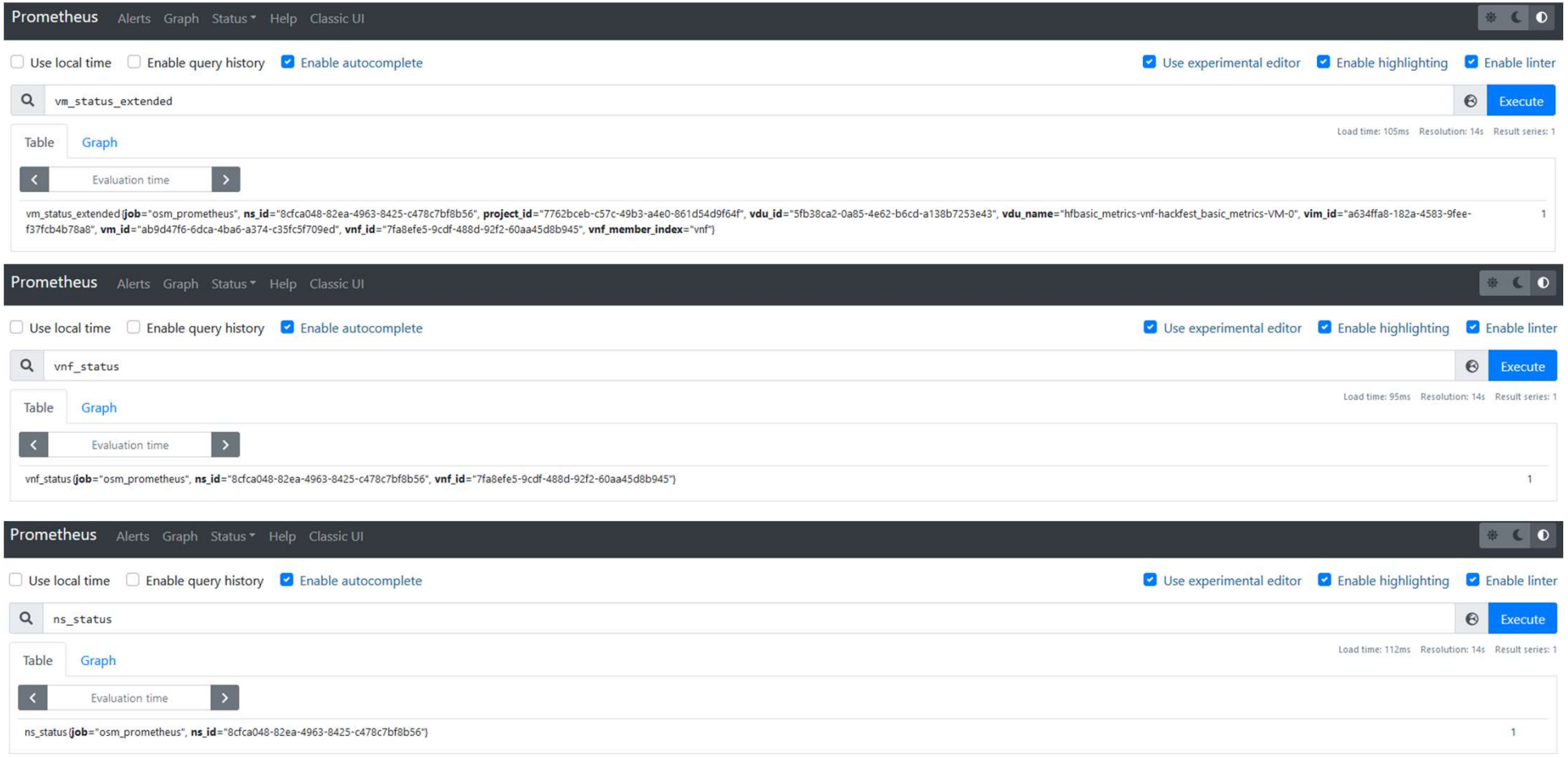


Metric derivation

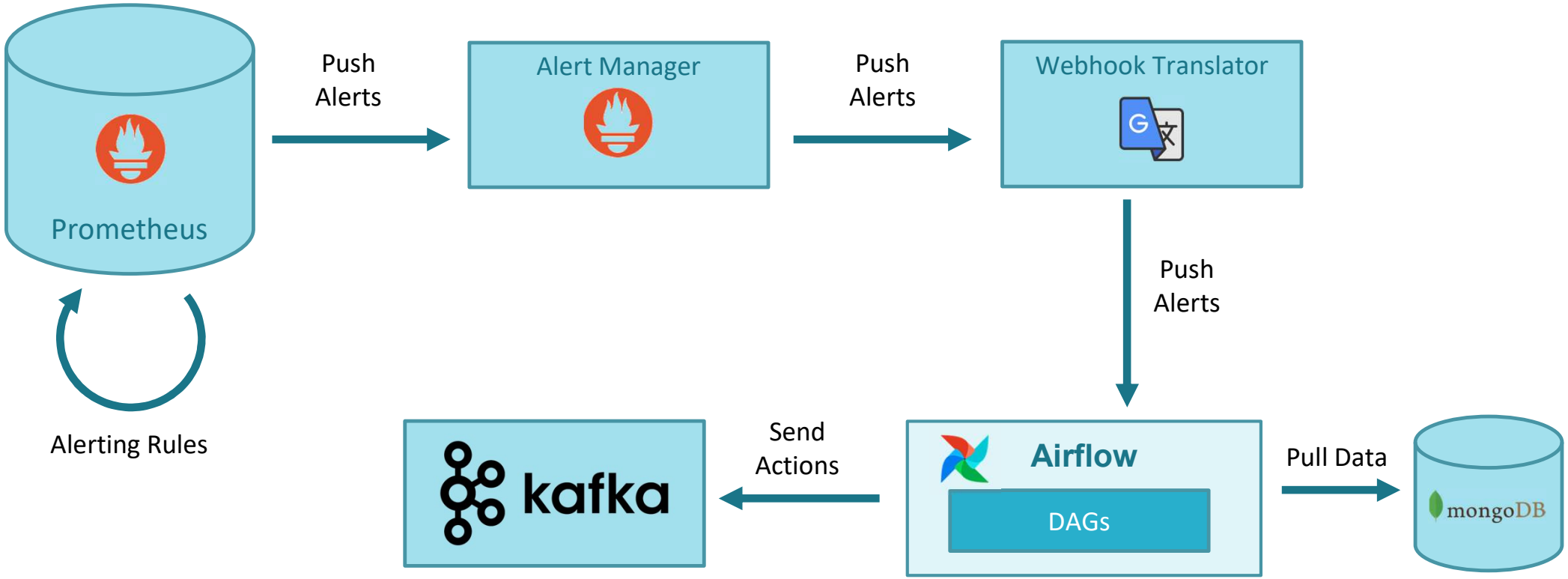
- Extended VM status:
  - From Prometheus (NS topology, VM status) to Prometheus
  - SW used: Prometheus Recording Rules
- VNF status:
  - From Prometheus (Extended VM status) to Prometheus
  - SW used: Prometheus Recording Rules
- NS status:
  - From Prometheus (Extended VM status) to Prometheus
  - SW used: Prometheus Recording Rules

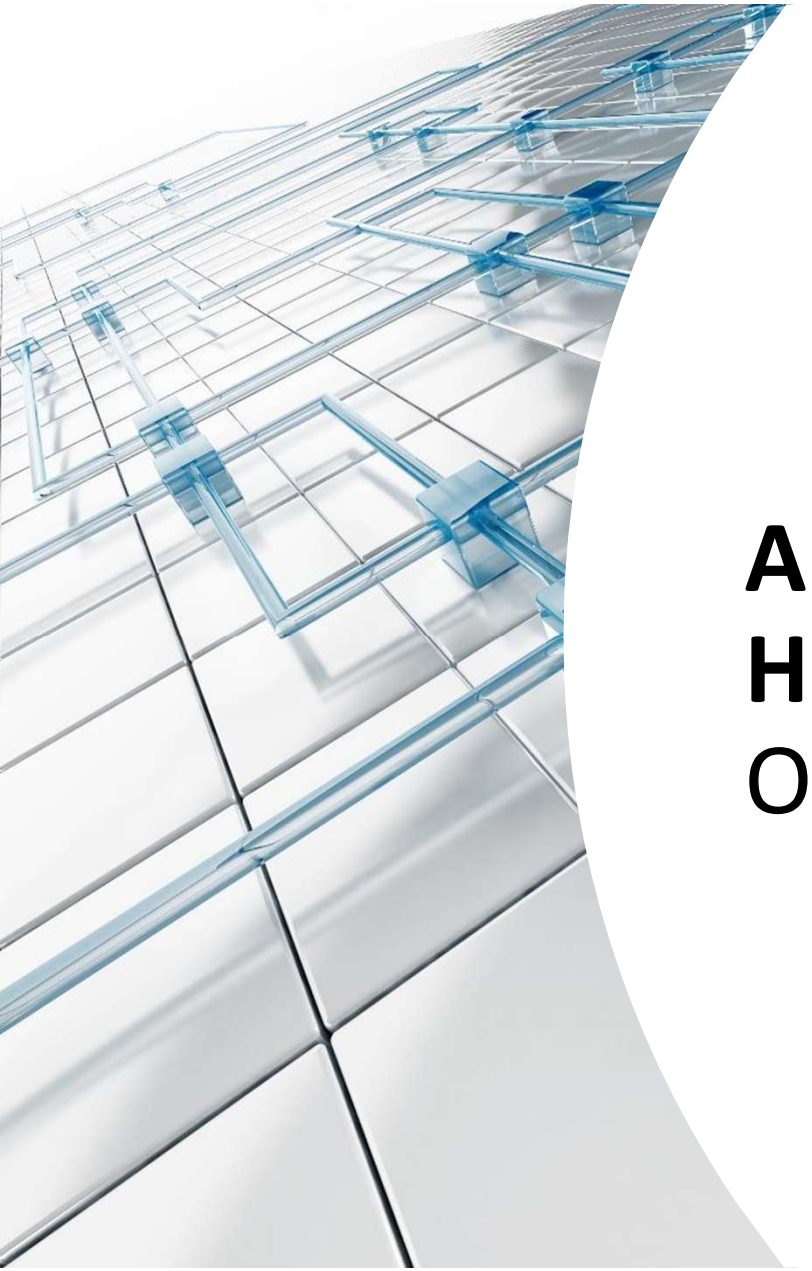


Screenshot of the derived metrics in Prometheus



Closed loops with new SA architecture





Open Source MANO

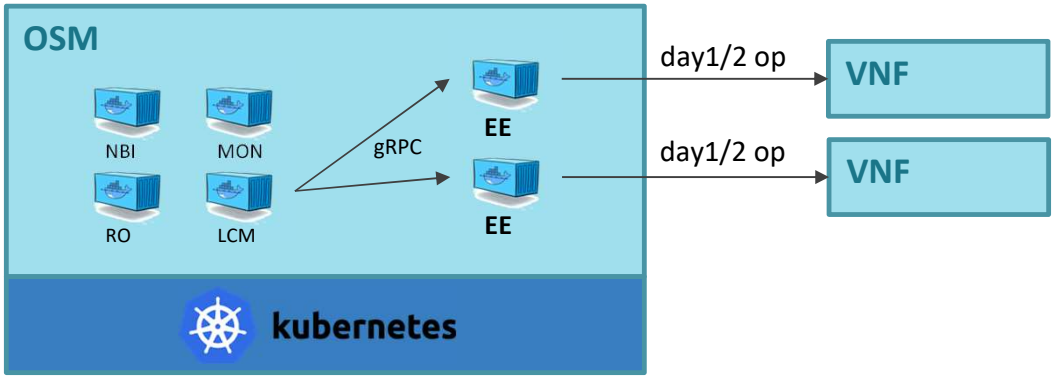
Authenticated gRPC for Helm-based EE

Overview and demo

Description

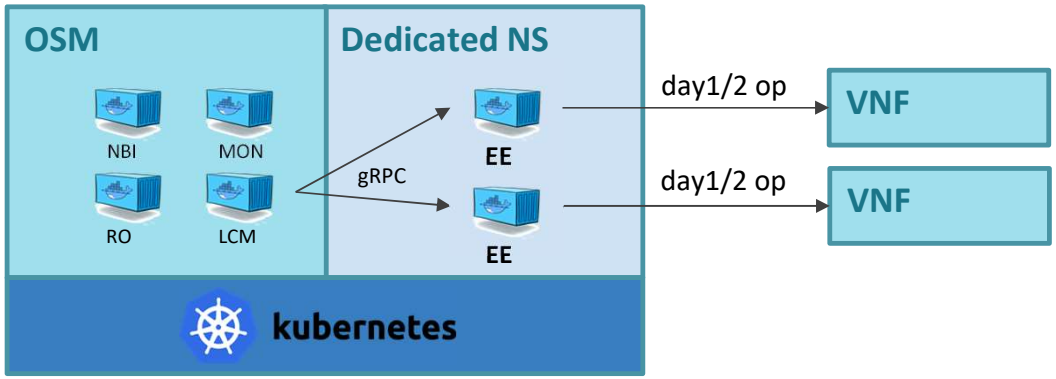
There were 3 vulnerabilities with Helm based Execution Environments (EE):

1. The EE were deployed in the same namespace than OSM, so it has access to all its secrets.
2. The gRPC Server in the EE pod received commands in plain text and from any client without authentication
3. The EE pod could have any capability (SYS_ADMIN), mount hostpaths, host network.



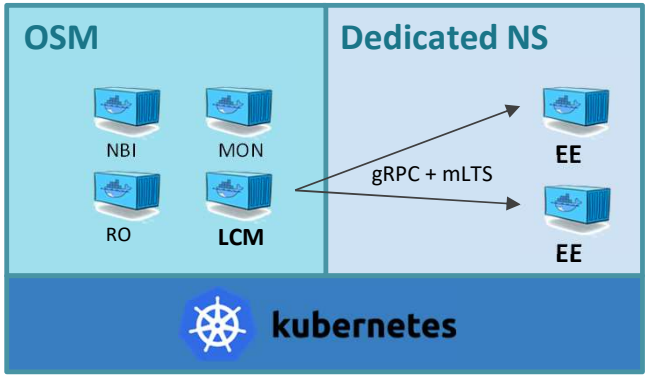
Isolated Execution Environments

- OSM now creates a dedicated namespace for each Network Service.
- All the Execution Environment data will be stored in its own namespace as secrets or configmaps
- RBAC allows reading objects in the same namespace
- The isolation does not interfere with the execution of primitives



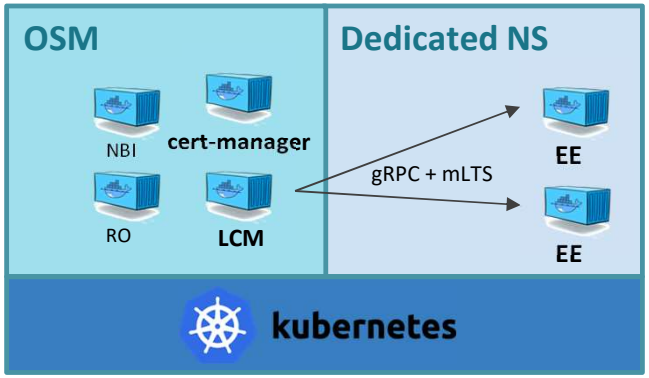
gRPC security

- Mutual TLS is implemented
  a. gRPC Client: LCM pod
  b. gRPC Server: EE pod
- On client side:
  a. LCM tries to establish a TLS connection to the EE pod. It verifies the Server's certificate with the CA
  b. All gRPC traffic is encrypted
- On the server side:
  a. EE pod requests client certificate
  b. Only LCM pod can sign a valid certificate and establish the connection.



gRPC security: Certificate management

- Since all gRPC communication is internal and the user has no control over it, self-signed certificates are used.
- *cert-manager* is used as the certificate management tool.
  - A self-signed Certificate Authority is created at the installation of OSM
  - A cluster issuer is created with the CA
  - The cluster issuer is used to generate Server certificates for each Network Service
  - The same cluster issuer generates the OSM client certificate



PodSecurity Admission policy

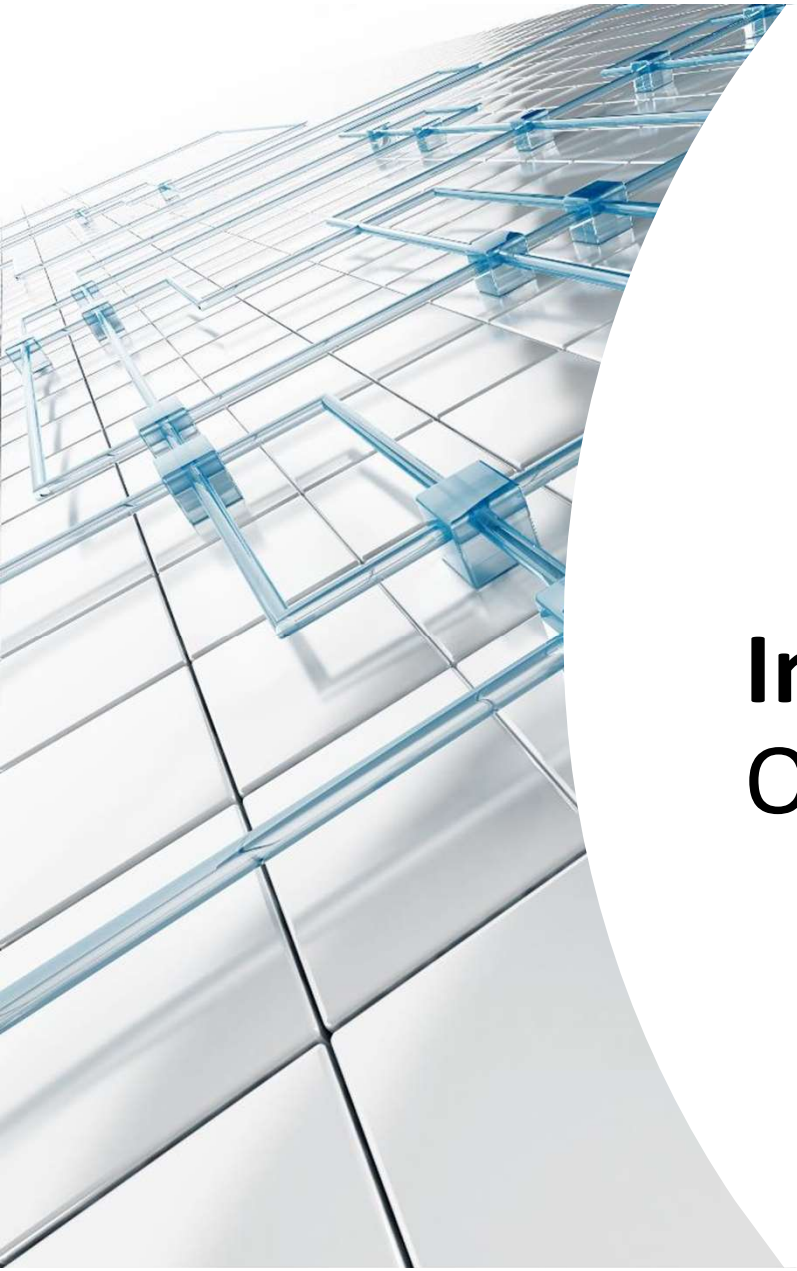
- We leverage the built-in *podSecurity* admission controller (available since k8s 1.23) to give Helm based EE's a "baseline" policy and prevent things such as privilege escalation, "dangerous" capabilities, etc., and still be able to run as *root*.

- Users can change the "baseline" policy to the stricter policy "restricted". This will prevent any root pod to be created in a global way:

```
kubectl patch cm -n osm osm-lcm-configmap -p '{"data":
{"OSMLCM_VCA_EEGRPC_POD_ADMISSION_POLICY": "restricted"}}'
```

Demo

- We will try to exploit the old vulnerabilities.
  - Try to impersonate LCM and send primitives to an EE
  - Try to read gRPC traffic between LCM and EE
  - Try to access secrets in OSM namespace from an EE pod
  - Try to create a privileged EE



Open Source MANO

Improved User Management
Overview and demo

Content

- User Management Overview
- User Management - OSM
- Security Implementation
- Advantages
- Demo

User Management Overview

- User Management is an important security requirement for any orchestrator application.
- It allows administrators to control and manage access to the application and its resources while ensuring security and compliance with industry best practices.
- Administrators need powerful user management capabilities to maintain user status and also define policies.

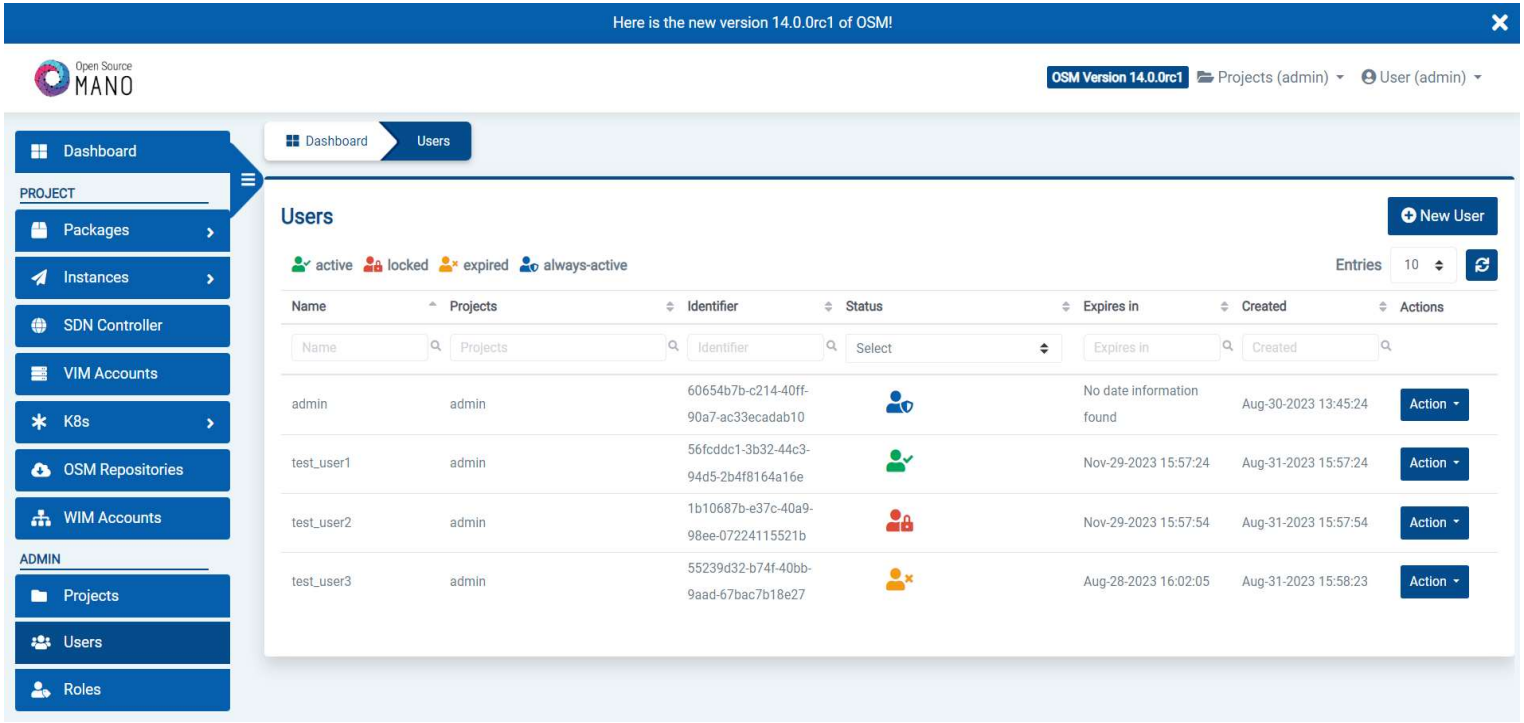
User Management in OSM Orchestrator

Some of the key considerations for user management are:

- **Authentication**: The first step in user management is to ensure that only authenticated users can access the application.
- **Authorization**: Once users are authenticated, the next step is to ensure that they have access to the appropriate resources within the application. Authorization is the process of defining what users can and cannot do within the application based on their roles, permissions, and privileges.
- **User Roles and Permissions**: User roles are used to define a set of permissions that a user has within the application. This can range from basic read-only access to full administrative privileges.
- **User Creation and Management**: The ability to create and manage users is an essential aspect of user management. Administrators should be able to create new users, modify existing user accounts, and delete users as necessary. Additionally, user accounts should be managed in a way that ensures security and prevents unauthorized access.

Security Implementation

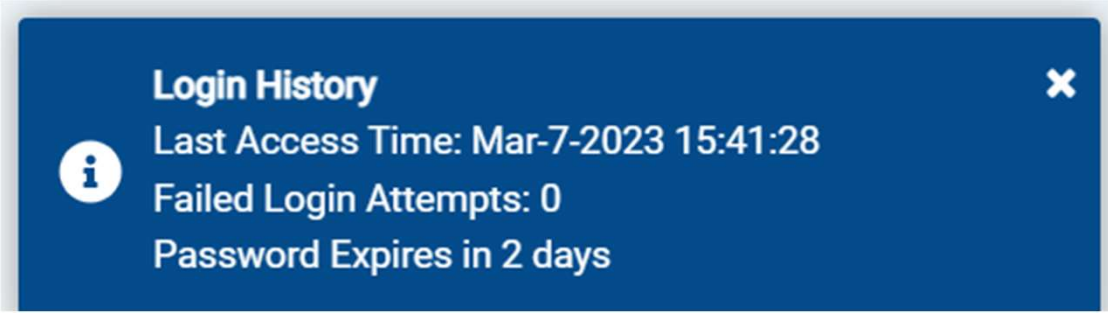
- **User Activity**: User activity is important for tracking user behavior and detecting potential security breaches. Application administrators should be able to review user status to identify suspicious activity and take appropriate action as needed.



Security Implementation

- **Password Policies**: Strong password policies can help prevent unauthorized access to the application. Password policies should include requirements for password length, complexity, and expiration. Additionally, users should be prompted to reset their passwords periodically to ensure continued security, and users should not be able to reuse the last 3 passwords.

- **Password Notification**: Password notification is a process of notifying users to change their passwords after a specified period.



Security Implementation

- **Account Renewal**: It is an important aspect of user management that ensures only authorized users have access to an application's resources. It is a process of extending the validity of a user account.

- **User Support**: User management should include support for users who experience issues with their accounts. This can include features such as password reset or account renewal.

Advantages

- **Improved Security:** Enforcing user login and password notifications helps to improve the security of an application by reducing the risk of unauthorized access.
- **Account Management:** Account expiration policies allow administrators to manage user accounts efficiently.
- **Centralized Management:** Enforcing these policies in user management provides a centralized approach to managing user accounts, making it easier for administrators to manage large numbers of users.
- Reduced Risk of Data Breaches.
- Password notifications remind users to update their passwords regularly, making it harder for attackers to crack weak or old passwords.

Demo

- Enforcing password change
- Locking user account on exceeding failed login attempts and performing unlock action
- Expiring the user account and performing renewal action
- Displaying the login history information in NGUI

Open Source MANO by ETSI

Thank You!

osm.etsi.org
osm.etsi.org/docs/user-guide
osm.etsi.org/wikipub