

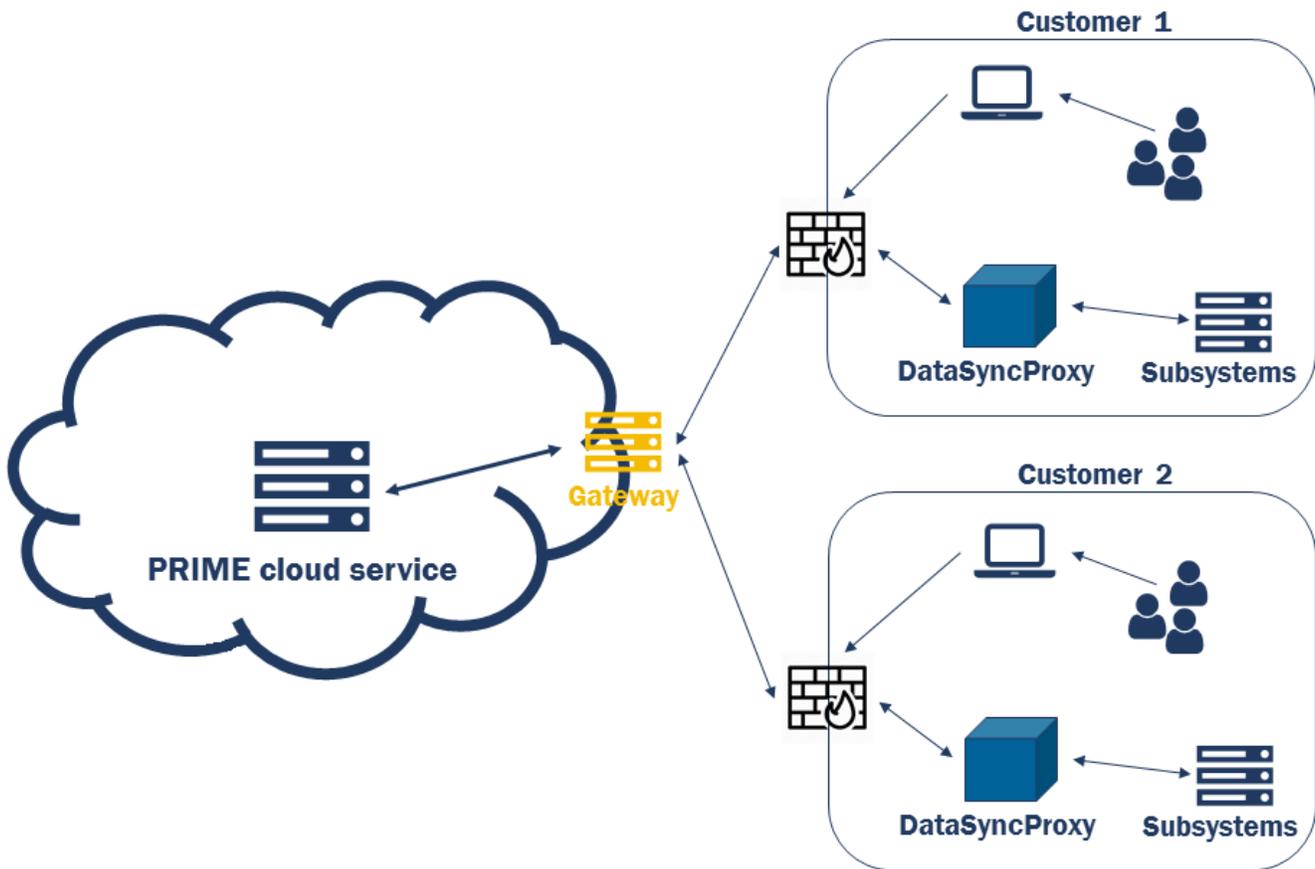
3.8 - Access local services from Nexus PRIME in the cloud

When PRIME is running in the cloud it does not have access to internal customer systems, like, for example, LDAP. It can not be guaranteed that a service in the customer's network can be accessed. To get access to customer systems, the DataSyncProxy is introduced. The DataSyncProxy is a small Spring Boot application. It is placed and runs on a machine in the customers network environment. The machine on which the DataSyncProxy runs must have access to internal systems like LDAP. The DataSyncProxy connects with PRIME (in the cloud) with a normal http connection. The connection is held as long as the proxy is running. While the connection is held, PRIME (in the cloud) is able to send Server-Sent Events (SSE) to the DataSyncProxy. The DataSyncProxy allows accessing a subsystem (search and export) in the same way as with any data source in a non-cloud installation of PRIME.

This article is valid from Nexus PRIME 3.8.

The picture illustrates how the DataSyncProxy is placed in the customer's network.

- Customer 1 and Customer 2 symbolizes networks at two different customers that connect to the PRIME cloud service via the gateway.
- When the user performs a search in PRIME, the DataSyncProxy provides a data connection to the user's subsystem and transfers the search result data to PRIME. Subsystems could be the LDAP, an external JDBC database, a SCIM data source, etc.
- The DataSyncProxy also allows the export of data from PRIME to the user's subsystem.



Connect the DataSyncProxy to PRIME

In PRIME's *system.properties* you can configure the following values. This has to be done for PRIME Explorer (for executing searches, exporting data, ...) and for PRIME Designer (for testing connection, getting the scim resource types and getting the attribute lists).

Example: Configure DataSyncProxy in system.properties

```
# turn DataSyncProxy on/off
dataSyncProxy.enabled=true
dataSync.responsePollingIntervalInMs=200
dataSync.responsePollingTimeoutInMs=30000
dataSync.proxyConnectionTimeoutInMs=3600000

# when the connection timeout above occurs, spring logs a message on error
level. To disable this enter: (Note: if you want to be sure not to miss
any other logs do not turn this on)
dataSync.suppressSpringMvcExceptionsWhileTimeout=true
```

In the *custom.properties* of the DataSyncProxy you can configure the following values:

Example: Configure DataSyncProxy in custom.properties

```
# Custom logging configuration
logging.config=./log4j2.xml
# The Url of the Prime server
prime.rest.url=https://123.456.789.121:8443/prime_explorer
# The Tenant id the Proxy is used for
prime.rest.tenantId=1
# The userName for which to call the Rest Service
prime.rest.userName=admin
# The password of the user which triggers the Rest Service
prime.rest.password=admin
# The delay how often the proxy checks if the connection is still active
(in milliseconds)
prime.rest.reconnect.delay=15000
```