# OpenDRAC Dynamic Light Paths

## *NOC Manual*


Author: André van der Ark


Version: 0.1

Date: 21-11-2011


SURFnet bv | Radboudkwartier 273 | Postbus 19035, 3501 DA Utrecht
T 030 - 2 305 305 | F 030 - 2 305 329 | admin@surfnet.nl | www.surfnet.nl

# Table of Contents

**1 Introduction ..... 3**

**2 OpenDRAC ..... 4**

2.1 Dynamic light paths ..... 4
2.2 Overview OpenDRAC ..... 4
2.3 Schedules and Services ..... 6
2.4 User Management ..... 6

**3 Use case: Request a dynamic light path ..... 8**

3.1 Step1: Determine port at customer site ..... 8
3.2 Step 2: Reserve the capacity towards centre stage ..... 8
3.3 Step 3: Physically install the port at the customer site ..... 9
3.4 Step 4: Enter port and capacity ..... 9
3.5 Step 5: Create the admin user group and resource group ..... 10
3.6 Step 6: Notify customer ..... 14

**4 Failure procedures ..... 16**

**5 Contact ..... 17**

# 1 Introduction

OpenDRAC is an open source project that aims to create a state-of-the-art piece of middle ware that allows network control by users and applications. It aims to be compatible with open standards, and where these don't exist it wants to be an appropriate proving ground.

The Dynamic Resource Allocation Controller, or OpenDRAC for short, is an implementation of a grid-like resource broker providing end application control of network resources. The primary goal of OpenDRAC is to expose network bandwidth to end user/application control while preventing unauthorized access and resource theft. OpenDRAC is capable of providing an application with bandwidth-on-demand, as well as guaranteed reservation of bandwidth for utilization at a later point in time. To this goal OpenDRAC offers the possibility to schedule services.

Originally conceived and prototyped back in 2000 by the Nortel Advanced Technology organization under the name CO2 (Content Over Optical), OpenDRAC was specifically designed to control hybrid Ethernet/optical networks for high-bandwidth scientific applications and storage area networks.

Production of OpenDRAC began in 2005 when SURFnet bv requested a deployable solution for their new SURFnet6 hybrid network. Development of the OpenDRAC product was then taken over by the Nortel Metro Ethernet Networks (MEN) organization, where support for the product continues today.

This document offers a practical guide for employees of the NOC. It starts with an explanation of some general configuration paradigms and a (simplified) architectural overview. For an extensive description of the architecture read the documentation at: https://svn.surfnet.nl/trac/openOpenDRAC/wiki/OpenDRACArchitecture.

After the chapter with general explanations is a chapter describing two flows. One for requesting a dynamic light path and one for allocating resources to users. The two flows use two clients of OpenDRAC: The web GUI and the Admin Console. Both clients have a dedicated manual for an in depth explanation of that client.

The last two chapters contain tips for problem solving and contact information for questions about the OpenDRAC project.

# 2 OpenDRAC

This chapter describes some general paradigms of OpenDRAC.

## 2.1 Dynamic light paths

Besides the standard routed IP-network SURFnet offers optical connections between two points. These are called light paths. Light paths offer high bandwidth (up to 10Gb/s) at high reliability. For one, this reliability is so high because a backup path is created for each light path. This backup takes over if the primary path fails.

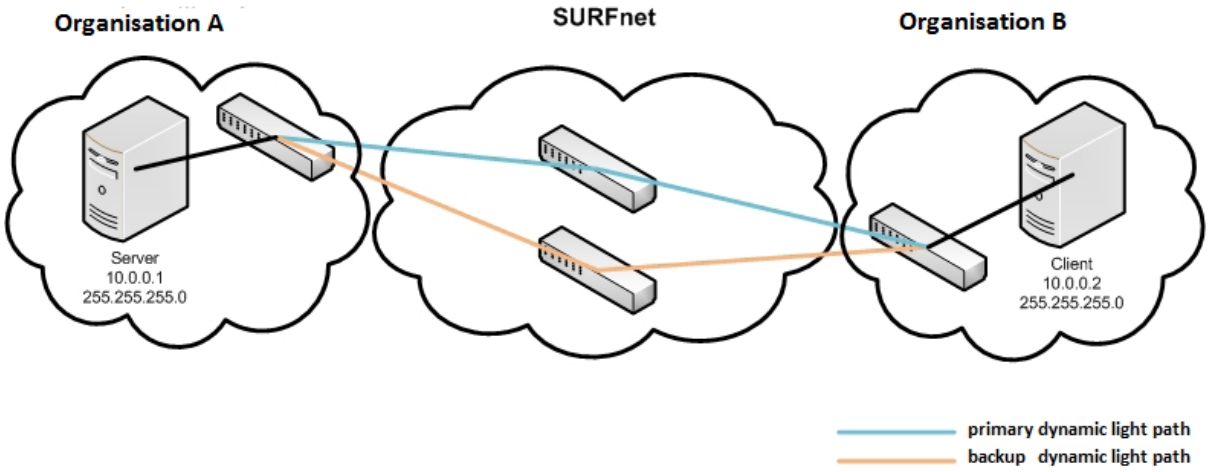


**Figure 1: Light path between two organisations**

SURFnet already uses static light paths in its network for high standard research tasks, but e.g. also for periodically creating backups. These light paths set up a connection between two points, e.g. two universities.

Dynamic light paths enable end users to temporarily set up a light path themselves. These paths can be with other users in the network of SURFnet. Once they've gotten the availability over resources, they can use these to their liking. This can be done with OpenDRAC.

## 2.2 Overview OpenDRAC

The image below depicts an overview of OpenDRAC and it components. At the core of the system is the server of OpenDRAC, the proxy and the database. This part controls the network elements through which the light paths have to be laid. In case of SURFnet these components are active in a Local Area Network (LAN).

The proxy communicates with Remote Method Invocation (RMI), a Java technique. The server communicates with the data base via Java Data Base Connectivity (JDBC). Lastly the server communicates with the network elements via Transaction Language 1 (TL1).

The proxy of OpenDRAC offers several interfaces for communication with outside software over the internet.

1. The web client: This is an HTTP interface that enables a client that runs in a web browser. The focus of this program lies in management of schedules and user management.
2. The Admin Console: This is a Java program that runs on the desktop of the user. The console uses RMI for communicating with the proxy. The focus of this program lies on populating the server with the proper values of the controlled network elements and on schedule management.
3. The web service client. This client consists of a series of soap. This is not so much a client as it is an interface. You need a separate program to configure and send the SOAP calls. (SOAP = *Simple Object Access Protocol)*
4. The Automation Tool: This is a little java program that is accessed via the command line of your operating system. It offers a simple interface to perform some basic actions like creating and cancelling a schedule. This program communicates with SOAP.

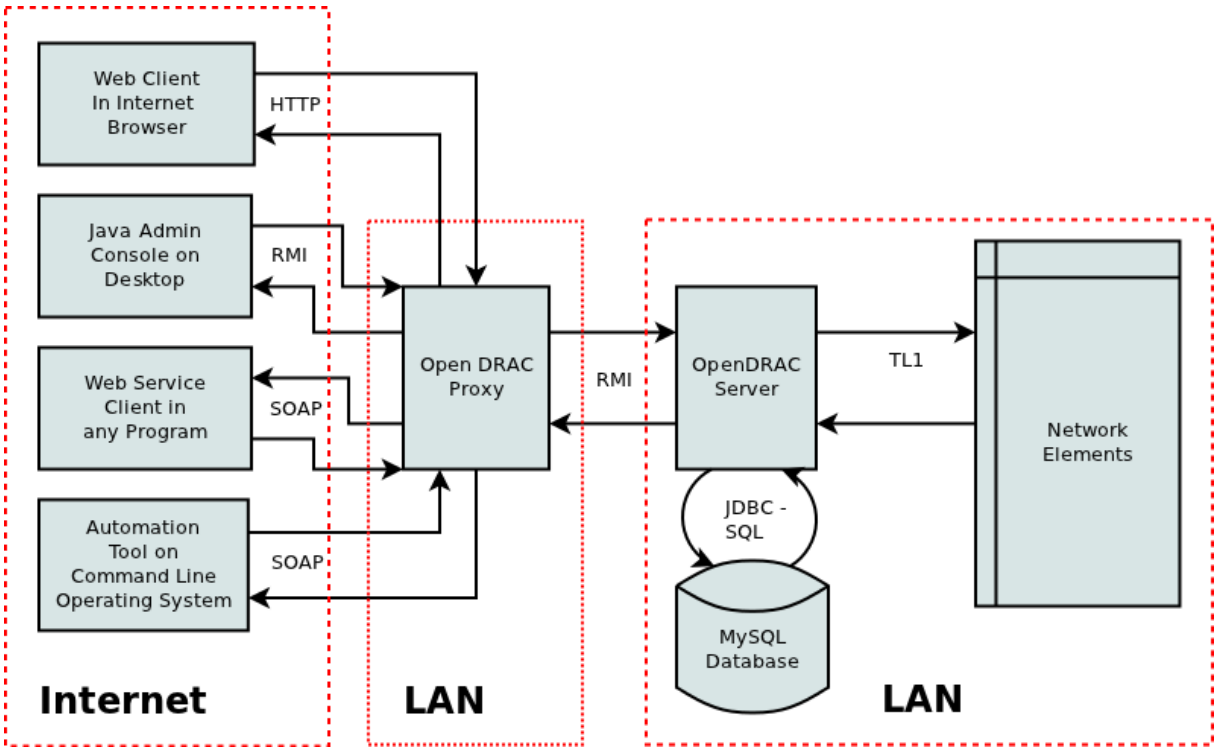


**Figure 2: Architecture Overview OpenDRAC**

For security reasons the proxy is separated from the server of OpenDRAC, each running in a different LAN.

## 2.3 Schedules and Services

The primary goal of OpenDRAC is to enable users to temporarily reserve bandwidth on dedicated "lines" between two end points. At creation of the reservation the path between the two end points is computated and it is this physical path that is claimed during execution of the reservation.

A reservation is made as a schedule. A schedule has a duration and can repeat periodically. E.g. you can create a schedule that reserves 1000Mbit/sec between two endpoints every Monday between 12:00h and 13:00h for the coming year.

When a schedule is created one or more services are made, depending on the number of repetitions. It is this service that makes the actual reservation on the network.

A service can be altered after creation, provided it's start date is in the future. Changing a service does not alter the schedule or any other service that is created with the schedule. It is also possible to create a new service using a schedule as a template. Such a service is then added to the list of services derived from the schedule.

## 2.4 User Management

OpenDRAC has an elaborate hierarchy for user management. This hierarchy consists of user groups, resource groups and users. The hierarchy is built up as follows:

- For an organization a parent resource group is set up. The parent resource group may contain zero or more sub resource groups.
- A resource group can contain zero or more user groups.
- A user group can contain zero or more users.

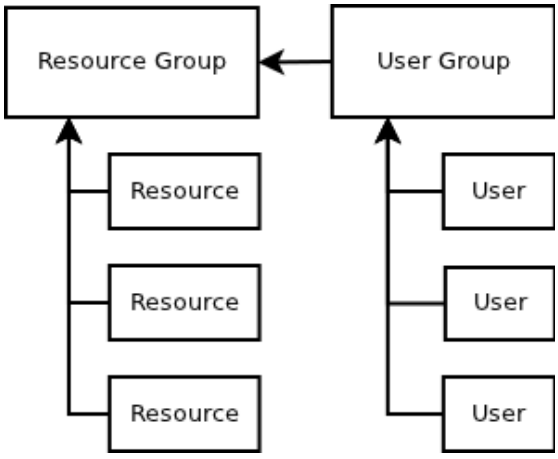


**Figure 3: Simple schematic resource management**

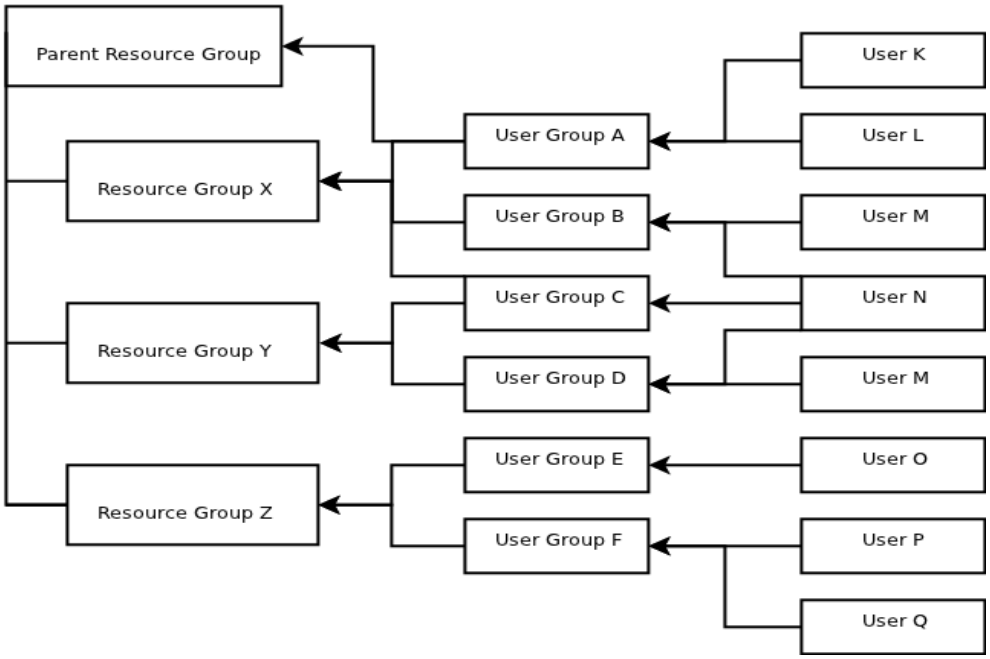


**Figure 4: Schematic representation of hierarchical structure resource management**

As depicted in the diagram above a user can be attached to multiple user groups and a user group can be attached to multiple resource groups.

To control the set of actions users can perform, multiple types of user groups haven are defined. There are three types of user groups:

1. User: Offers access to basic user capabilities through the system.
2. System Administrator: Offers access to all capabilities through the system.
3. Group Administrator: Offers access to basic user and group administrator capabilities through the system.

# 3 Use case: Request a dynamic light path

The image below shows the numbered steps to request a dynamic light path (DLP) . these steps are described in the following paragraphs.

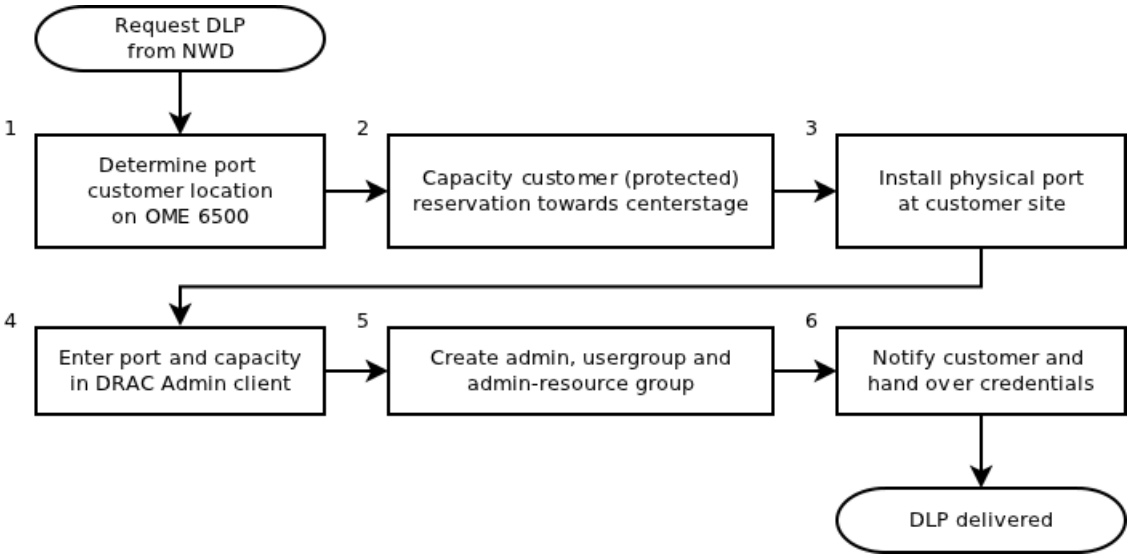


**Figure 5: Steps for requesting a dynamic light path**

## 3.1 Step1: Determine port at customer site

This is done the same as for requesting a static light path. You'll have to factor in the protected path on the port. The customer has registered the request in the control plane.

## 3.2 Step 2: Reserve the capacity towards centre stage

In and around the centre stage capacity has been assigned to DRAC. The planning tool can display an overview of the claimed capacity for OpenDRAC. The customer site needs to obtain a protected 1Gbit/s line towards (the nearest) OME 6500 that is already assigned capacity.

1. Determine the nearest OME6500 network element to which OpenDRAC capacity is assigned.
2. Claim, by using the planning tool, protected capacity between the customer site and the location of step 1. This capacity is required in step 4 for claiming in the Admin Client.

## 3.3 Step 3: Physically install the port at the customer site

The installation of the customer port (SPF) on the OME6500 of the customer is a standard installation, just like the installation of a port for a static light path. The exact port location is required in step 4 to enter the port in the Admin Client of OpenDRAC.

## 3.4 Step 4: Enter port and capacity

This step is described in two paragraphs: one for the port and the other for the capacity.

### 3.4.1 Step 4a: Enter the port in the Admin Client of OpenDRAC

1. Start the admin Client and log in as admin
2. If it concerns a new network element (not registered in OpenDRAC), right click in the left pane and choose "Add NE". User name is 'DRAC01'. enter the IP-address and password
3. In tab facilities click on the button "Retrieve"
4. Using the naming scheme for the NE look for the right network element and select the port of step 3. Right click the selected line and choose "Edit".
5. Click on "Generate" and the port name is filled in according to standard naming in OpenDRAC.
6. Set the signalling type to UNI and click on "Ok". The port is now available in the web client and can be assigned to uses by the admin.

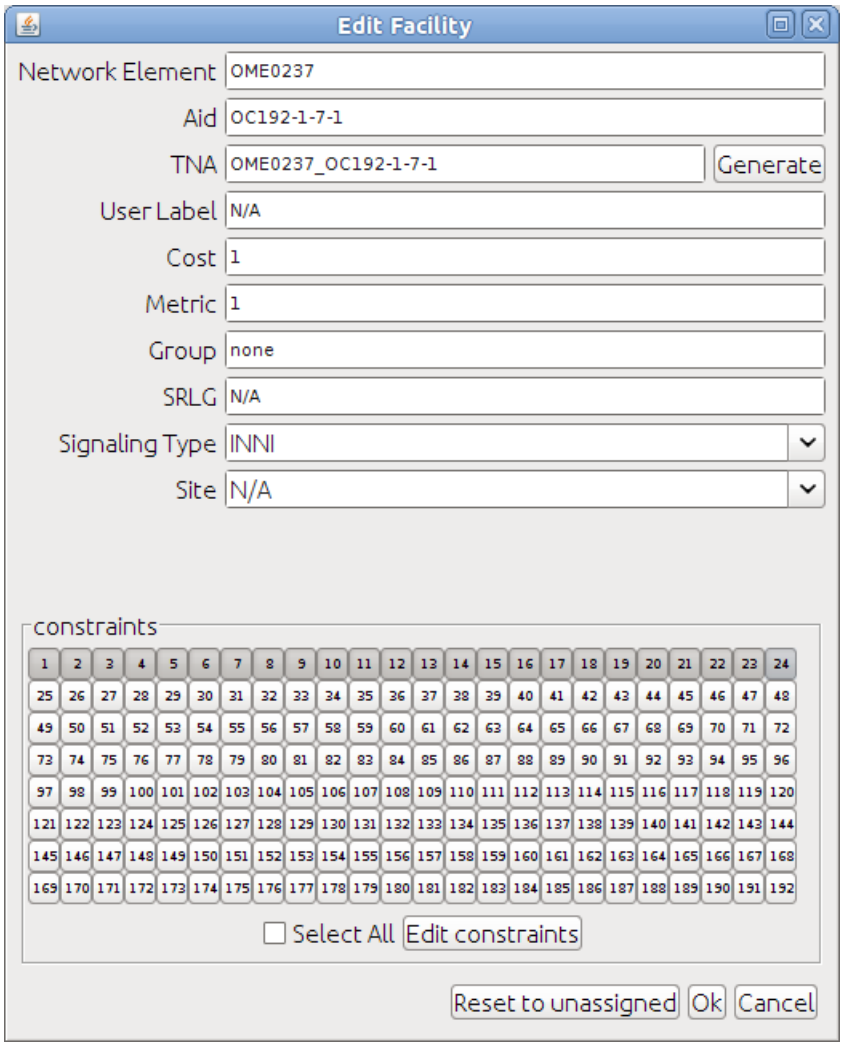


**Figure 6: Editing port definition**

### 3.4.2 Step 4b: Enter capacity in the Admin Client

1. Click on the tab "Facilities" in the Admin client and select the series of reserved timeslots of step 2
2. Right click on the port
3. Click on "Generate" to assign the name
4. Set the port type to INNI
5. Select "Edit constraints". A warning is displayed. Assign the time slots. Light grey means assigned to OpenDRAC. Dark grey means not assigned to OpenDRAC. Click on OK. The port can now facilitate the chosen time slots for a dynamic light path.
6. Repeat step 4b until all time slots (protected) from step 2b of the process are executed.

## 3.5 Step 5: Create the admin user group and resource group

Log in as administrator in the web client and execute the 3 steps described in the following paragraphs. The web interface is accessible through: **drac.surfnet.nl**

The three sub steps are:

1. Create/edit Admin resource group
2. Create Admin user group
3. Create Admin user

### 3.5.1 Step 5a: Create/edit Admin resource group

1. See if the resource group already exists, or if it needs to be created. Go to Security → Resource Group Management → List/Edit Resource Groups; click on 'Submit'. Click on the requested resource group if it is displayed in the list. Select Security → Resource Group Management → Create Resource Group otherwise.
2. If the resource group already exists, assign the created port and click on 'Submit.'
3. If the resource group does not exist, the screen below appears.
4. At 'Name' fill in the customer name , followed by a hyphen (-) and the word 'AdminResourceGroup'.
5. Assign at the tab 'Resources' the port(s) of this request to the resource group.
6. Click on 'Submit'. De customer-admin-resource group is now created. This group is required in step 5b to assign the  de customer-admin-user group.

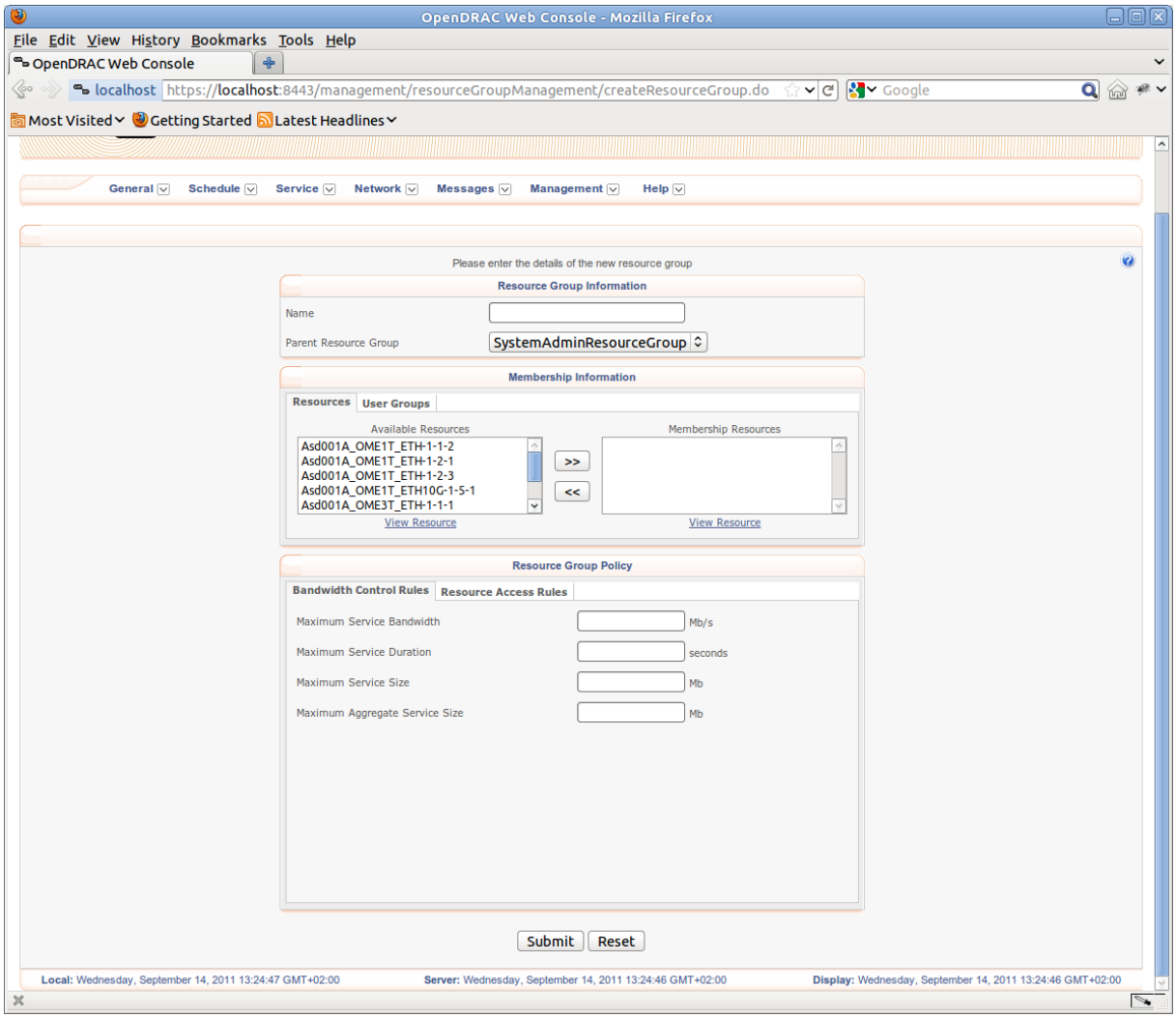


**Figure 7: Editing admin resource group**

### 3.5.2 Step 5b: Create Admin-user group

1. See if the admin-user group already exists, or if it needs to be created. Go to Security → User Group Management → List/Edit User Groups; click on 'Submit'. Click on the requested user group if it is displayed in the list. If it is not in this list, go to Security → User Group Management → Create User Group.
2. If the user group already exists, assign the created resource Group and click on 'Submit.'
3. If the user group does not exist, the screen below appears.
4. At 'Name' fill in the customer name , followed by a hyphen (-) and the word 'AdminGroup'. Set the 'Group Type' to 'Group Administrator'. As 'Parent User Group' select 'SystemAdminGroup'.
5. Under tab 'Resource Groups' assign the admin-resource group from step 5a to this user group.
6. Click on 'Submit'. The customer-admin-user group is now created. This group is required to assign a customer-admin user in step 5c.

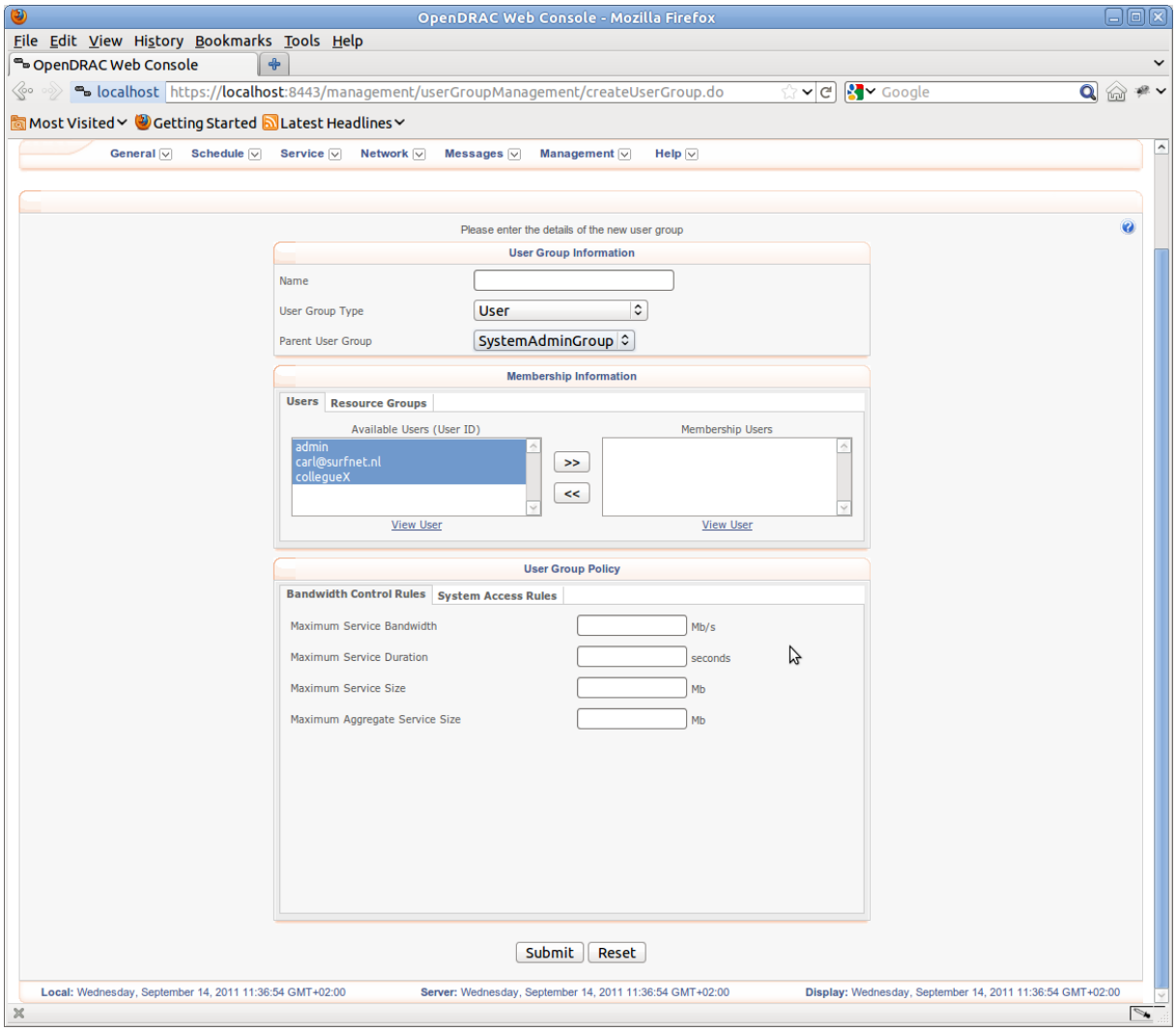


**Figure 8: Editing user group**

### 3.5.3 Step 5c: Create Admin-user (for the customer)

1. Execute step 5c completely if in step 5b a new admin-user group is created. If the customer already has an admin-user, proceed with step 6.
2. Go to 'Security → User Management → Create User'. the following screen appears.
3. Set for name the customer name, followed by '-admin-' and possibly an extra digit or text post fix.
4. Leave account state to Enabled and Authentication Type to Internal.
5. Enter the password thrice; WS Password can be used by the customer to get access to the DRAC-API.
6. In tab 'User Group Membership' assign this user to the customer-admin-user-group of step 5b.
7. Click on 'Submit'. The customer-admin-user is now created, belongs to a customer-admin-user-group, and contains at least one resource.

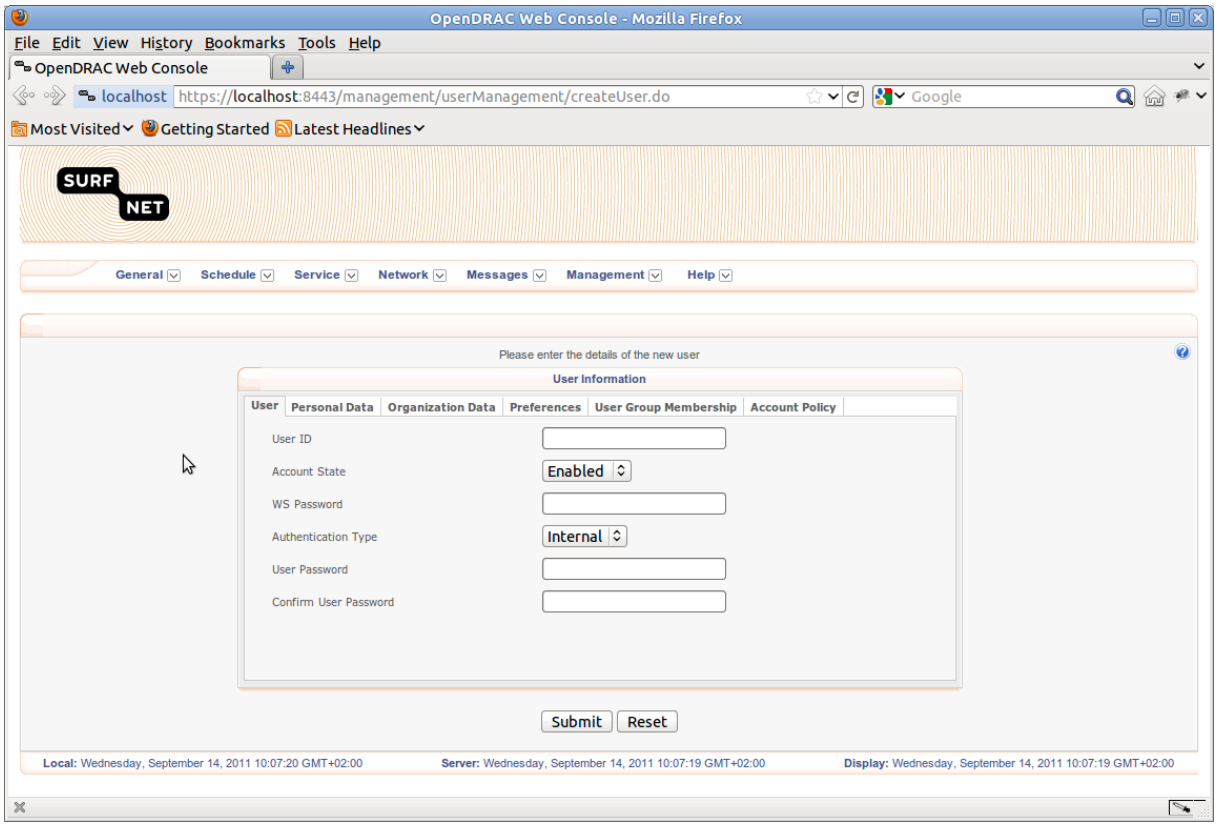


**Figure 9: Editing user**

## 3.6 Step 6: Notify customer

Send an **encrypted** mail to the customer, template:

Dear <>,

In this mail I send you the requested data for the requested light path in the SURFnet network

At the SURFnet NOC we welcome questions and remarks.

Location/port name: Xx000A_ODF01_1

Control plane: MM fibre

URL Maintenance dynamic light paths: drac.surfnet.nl

Admin-account customer-admin-post fix

Admin-password xxxxxxxx

With kind regards,

SURFnet NOC

+31 (0)30 2477586

# 4 Failure procedures

In the event OpenDRAC does not function as expected, there are several things a user can do to find information about the cause:

- Login to the web-interface drac.surfnet.nl (no errors after log in?)
  - Check: General → View Logs: For filtering, multiple parameters are available
  - Check: assigning right UNI's to the right resource groups and users
  - Check: schedule as NOC-admin the path with the problem.
- Error message? Login to the Admin Client en check:
  - Assigning the right bandwidth between NE's
  - Assigning the right UNI's to OpenDRAC
- Ping each server from a shell.
- Login to servers via SSH. On MS-Windows you can do this with putty and on Linux, Unix or Apple this can be done from the command line in a shell.
  - Request the status of OpenDRAC via ~/DRAC_xxxx/bin/DRACcontrol status
  - Possibly resume the controller and proxy servers ./DRACcontrol halt ./DRACcontrol start
  - (!) The order is important: first restart the controller(s), then, after waiting 5 minutes, restart the proxy(s)

OpenDRAC can be configured to send emails for some special error situations, like a switch over to the backup instance of OpenDRAC, locking of users, sessions or IP-addresses due to multiple bad login attempts and shutdown of the server. These messages are formatted loggings of software sent as an email. The addressees (configurable) that receive such a mail can use the forementioned failure procedures to retrieve more information about the problem.

The example below shows a mail of a server shutdown:

```
LS,

OpenDRAC reported the following problem: 'Shutdown hook invoked, reversing
processes list'
Timestamp: 28 Oct 2011 11:16:43
Location in code: org.opendrac.launcher.Launcher$3.run(), line number:86
Priority of the event: WARN

This is an automated mail to which you cannot reply
```

# 5 Contact

SURFnet:

- Technical Product Manager Hans Trompert (Hans.Trompert@SURFnet.nl)
- Technical Product Manager Gerben van Malenstein (Gerben.vanMalenstein@SURFnet.nl)