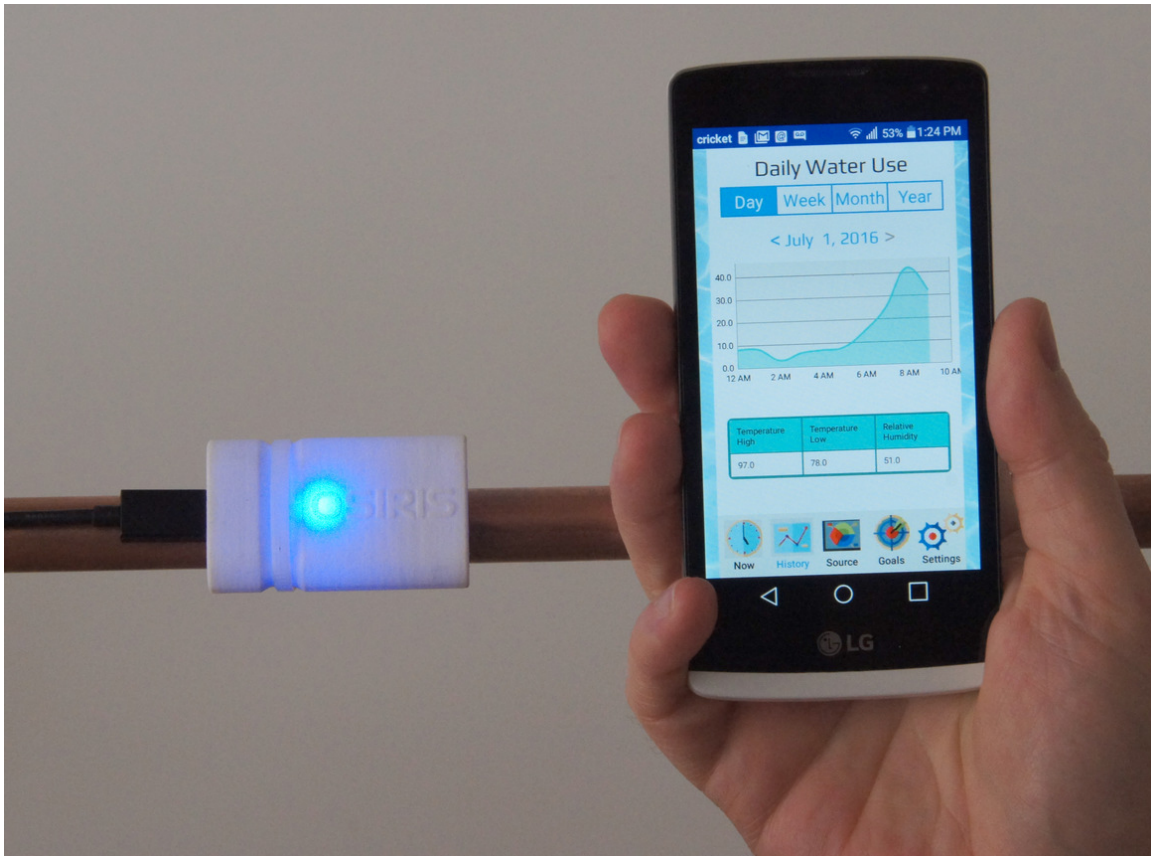


# OSIRIS

## User APIs



Rev: 1.0  
Published: Aug 24, 2018



# Table of Contents

<b>TABLE OF CONTENTS .....</b>	<b>2</b>
<b>LIST OF TABLES .....</b>	<b>3</b>
<b>1 BASICS .....</b>	<b>3</b>
1.1. INTRODUCTION .....	3
<b>2 SETUP.....</b>	<b>3</b>
2.1. SETUP OSIRIS IN YOUR HOME.....	4
2.2. ACCESS VIA CLOUD APIS .....	4
<b>3 APIS .....</b>	<b>4</b>
3.1. GET UNITS AND CONTACTS .....	5
3.1.1. <i>Requirements</i> .....	5
3.1.2. <i>Request</i> .....	5
3.1.3. <i>Response</i> .....	6
3.2. GET UNIT SETTINGS.....	8
3.2.1. <i>Requirements</i> .....	8
3.2.2. <i>Request</i> .....	8
3.2.3. <i>Response</i> .....	8
3.3. GET SENSOR DATA & GOALS .....	9
3.3.1. <i>Requirements</i> .....	10
3.3.2. <i>Request</i> .....	10
3.3.3. <i>Response</i> .....	11
3.4. I'M GETTING ALARM.....	13
3.4.1. <i>Requirements</i> .....	13
3.4.2. <i>Request</i> .....	13
3.4.3. <i>Response</i> .....	13
3.5. MUTE ALARM.....	14
3.5.1. <i>Requirements</i> .....	14
3.5.2. <i>Request</i> .....	14
3.5.3. <i>Response</i> .....	14
3.6. CANCEL ALARM .....	14
3.6.1. <i>Requirements</i> .....	14
3.6.2. <i>Request</i> .....	15
3.6.3. <i>Response</i> .....	15
3.7. SET WATER VALVE .....	15
3.7.1. <i>Requirements</i> .....	15
3.7.2. <i>Request</i> .....	15
3.7.3. <i>Response</i> .....	15
3.8. SET GOAL .....	16
3.8.1. <i>Requirements</i> .....	16
3.8.2. <i>Request</i> .....	16
3.8.3. <i>Response</i> .....	16
3.9. SET UNIT SETTINGS .....	16
3.9.1. <i>Requirements</i> .....	16

3.9.2.	<i>Request</i> .....	17
3.9.3.	<i>Response</i> .....	17

## List of Tables

Table 1-	Access Level.....	5
Table 2-	User Fields.....	7
Table 3-	Unit Fields .....	7
Table 4-	Contact Fields.....	8
Table 5-	Get Unit Settings Args.....	8
Table 6-	Unit Settings .....	9
Table 7-	Recommended Plumbers .....	9
Table 8-	Get Sensor Data Args .....	11
Table 9-	Sensor Data & Goal Fields.....	13
Table 10-	I'm Getting It Args .....	13
Table 11-	Mute Args.....	14
Table 12-	Cancel Alarm Args.....	15
Table 13-	Set Goal Args .....	16
Table 14-	Set Unit Settings Args .....	17

## 1 Basics

### 1.1. Introduction

In addition to using the Android & iOS phone Apps, you may wish to integrate Osiris into your Smarthome or use with your Raspberry Pi or PC. To this end, AZ APPS offers a secure cloud-to-cloud interface to our server. You can access all of the same units that you can access in the App, without any need to change their firmware or connect to them directly. This allows Osiris to seamlessly interface to dozens of Smarthome solutions. Use it however you want, with your home setup of choice.

Simply use your HTTPS client to contact the AZ APPS server and submit your user info and REST request. You can request data or change device settings. All requests are formatted as JSON with UTF-8 encoding.

The APIs support Osiris Smart Water Monitor, Trident Smart Valve, and more devices coming soon.

## 2 Setup

## 2.1. *Setup Osiris in Your Home*

Install Osiris in any single-family home, on pipes of any size, in any country in the world. For best performance, please mount Osiris on a cold water pipe.

- 1) Mount Osiris on a pipe inside your home
- 2) Connect the power cable
- 3) Use the Android or iOS App to:
  - a. register Osiris
  - b. connect Osiris to your home WiFi network
  - c. calibrate your unit
  - d. setup your API email and password

The App currently does not allow you to change your API password. Resubmitting your API email with a different password will not change it. If you lose your password, you must submit a new email address and password.

## 2.2. *Access via Cloud APIs*

Osiris uploads data and downloads user settings from the AZ APPS cloud every 15 minutes. Osiris emails you whenever you have an alarm, someone responds to an alarm, or someone cancels an alarm.

The APIs are rate-limited to a number of calls per hour or day. The limits should easily give you access more often than you normally need. **Please do not attempt to access the APIs more times per hour than the rate limits.** If you do this on a regular basis, we will disable your API access. If this happens, you are welcome to contact us about it. If you simply made a mistake in your use of the APIs, we will be happy to reinstate your access.

The APIs give you almost as much access to your units as the phone Apps do. Manage alarms, graph water use, set goals, turn off the water, and more. Fully integrate Osiris and Trident into your preferred Smarthome applications.

## 3 APIs

You must submit your API email and password in every request. You may also need to submit additional fields like unit number.

The APIs give you the same level of access to each unit that the phone App gives you. For example, if you are the Customer who registered the unit, you can access everything. But if you are an emergency contact, you can only access the alarm and status pages of that unit in the App. Hence you can only access the alarm and status APIs for that unit. If you want more access, talk to the customer who owns that unit. It is up to them, not up to AZ APPS, who has access to their unit.

*Table 1- Access Level*

Access	Description	APIs
E	Emergency Contact	alarms, status
U	User	alarms, mute, status, water data, goals
F	Full Access	all
C	Customer	all

To prevent denial-of-service issues APIs have a maximum number of uses per user per hour. In other words, you may not make the same API call more times per hour than the rate limit.

You can see an example request and example response for each API.

**For the examples to work, you must use your API username & password, have curl installed, and your version of curl must support the --tlsv1 flag.**

Every correctly formatted request will return the 200 OK status, followed by details. Only the first example will show the 200 OK.

### **3.1. Get Units and Contacts**

Get your user info, the units you have access to, and your contacts who also have access to these units.

#### **3.1.1. Requirements**

Field	Requirement
Device	none
Access Level	emergency, user, full, customer
Rate Limit	5 requests per hour

#### **3.1.2. Request**

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure":
"Get_My_Units_And_Contacts" }' -H "Content-Type: application/json"
```

### 3.1.3. Response

```
HTTP/1.1 200 OK
Date: Thu, 23 Aug 2018 23:48:18 GMT
Server: Apache
X-Frame-Options: SAMEORIGIN
X-Powered-By: PHP/7.0.23
Vary: Accept-Encoding
Content-Length: 1303
Content-Type: text/html; charset=UTF-8
```

```
[
  [
    {
      "first_name": "Nick",
      "last_name": "Olivier"
    }
  ],
  [
    {
      "unit_id": 773,
      "model": "Trident",
      "unit_name": "Andre Home",
      "real_name": "Andre Olivier - Home",
      "permission": "U",
      "mute_minutes": 0,
      "mute_time": "2018-08-23 21:41:54",
      "is_alarming": "N",
      "alarm_type": null,
      "alarm_response": null,
      "trident_id": null,
      "flow_valve": "on"
    },
    {
      "unit_id": 774,
      "model": "Osiris",
      "unit_name": "Home",
      "real_name": "Osiris 774",
      "permission": "C",
      "mute_minutes": 0,
      "mute_time": "2018-08-23 21:45:54",
```

```

    "is_alarming": "N",
    "alarm_type": null,
    "alarm_response": null,
    "trident_id": null,
    "flow_valve": null
  }
],
[
  {
    "unit_id": 773,
    "first_name": "Andre",
    "last_name": "Olivier",
    "alarm_response": null
  },
  {
    "unit_id": 774,
    "first_name": "Bill",
    "last_name": "Dopson",
    "alarm_response": null
  }
]
]

```

*Table 2- User Fields*

Field	Description
First Name	first name
Last Name	last name

*Table 3- Unit Fields*

Field	Description
Unit Id	use in subsequent requests to this unit
Model	Osiris, Trident
Unit Name	what you called your unit in the App
Real Name	mode & unit if you are customer, else customer's name for unit
Permission	E = emergency contact, U = user, F = full, C = customer
Mute Minutes	minutes until mute expires, starting at mute time
Mute Time	UTC time when unit was muted
Is Alarming	M = muted, Y = alarming, N = not alarming
Alarm Type	hot, cold, flow, null
Alarm Response	Y = getting it, N = can't get it, null = no alarm
Trident Id	id of Trident paired with this unit
Flow Valve	on = water can flow, off = no water can flow, null = not trident

Table 4- Contact Fields

Field	Description
Unit Id	use in subsequent requests to this unit
First Name	first name
Last Name	last name
Alarm Response	Y = getting it, N = can't get it, null = no alarm

## 3.2. Get Unit Settings

Get a list of settings for your unit.

Get recommended plumbers in your unit's local area, that have registered with AZ APPS.

### 3.2.1. Requirements

Field	Requirement
Device	Osiris, Trident
Access Level	user, full, customer
Rate Limit	5 requests per hour

### 3.2.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure": "Get_Config",
"args": "774" }' -H "Content-Type: application/json"
```

Table 5- Get Unit Settings Args

Field	Description
Unit Id	unit id

### 3.2.3. Response

```
[
  [
    {
      "field": "appDisplayTempUnit",
      "val": "F"
    },
    {
      "field": "devAlarmColdC",
      "val": "4"
    }
  ]
]
```



```

    {
      "field": "devAlarmFlowMin",
      "val": "20"
    },
    {
      "field": "devAlarmHotC",
      "val": "54"
    },
    {
      "field": "devAlarmTempMin",
      "val": "30"
    },
    {
      "field": "devFlowOnPercentOff",
      "val": "115"
    }
  ],
  [
    {
      "Null": "Null"
    }
  ]
]

```

*Table 6- Unit Settings*

Field	Description
appDisplayTempUnit	display App temperatures in Fahrenheit / Celsius
devAlarmColdC	alarm when temperature < cold C
devAlarmFlowMin	alarm after detecting n minutes of continuous flow
devAlarmHotC	alarm when temperature > hot C
devAlarmTempMin	alarm after detecting n minutes of hot / cold temperature
devFlowOnPercentOff	water is on when flow level > percent * off level

Trident does not have flow settings.

*Table 7- Recommended Plumbers*

Field	Description
Name	business name
Slogan	advertising phrase
Phone Number	phone number

### **3.3. Get Sensor Data & Goals**

Get weather forecast, monthly goals, month data, day data, hour data, and paired Trident hour data for your Osiris.

Procedure	Description
Get_Full_Data	get all data for this Osiris, with Trident data truncated to last 3 days
Get_Recent_Data	same as full data, but Osiris hour data is truncated to last 2 weeks

Or

Get weather forecast and hour data for your Trident.

Procedure	Description
Get_Full_Data	get Trident data for last 3 days
Get_Recent_Data	get Trident data for last 3 days

Full gives you all available history of your unit. Download on rare occasions when your users want in-depth analysis of their water history.

Recent gives you most of the same data, but limits your hour-by-hour data to the last two weeks. This reduces the message size by up to 95%. Download to display water use to your users.

### 3.3.1. Requirements

Field	Requirement
Device	Osiris, Trident
Access Level	user, full, customer
Rate Limit - Full	2 requests per day
Rate Limit - Recent	3 requests per hour

### 3.3.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure":
"Get_Full_Sensor_Data_And_Goals", "args": "774" }' -H "Content-Type:
application/json"
```

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure":
"Get_Recent_Sensor_Data_And_Goals", "args": "774" }' -H "Content-Type:
application/json"
```

Table 8- Get Sensor Data Args

Field	Description
Unit Id	unit id

### 3.3.3. Response

```
[
  [
    {
      "forecast": "Sunny skies"
    }
  ],
  [
    {
      "local_date": "2018-08-01",
      "minutes": 0
    }
  ],
  [
    {
      "local_date": "2018-08-01",
      "flow_minutes": 0,
      "pipe_temperature_avg_C": 0,
      "temperature_low_C": 0,
      "temperature_high_C": 0,
      "temperature_avg_C": 0,
      "humidity_avg": 0
    }
  ],
  [
    {
      "local_date": "2018-08-21",
      "flow_minutes": 0,
      "pipe_temperature_avg_C": 0,
      "temperature_low_C": 0,
      "temperature_high_C": 0,
      "temperature_avg_C": 0,
      "humidity_avg": 0
    },
    {
      "local_date": "2018-08-22",
      "flow_minutes": 0,
      "pipe_temperature_avg_C": 0,
      "temperature_low_C": 0,
      "temperature_high_C": 0,
    }
  ]
]
```

```
    "temperature_avg_C": 0,  
    "humidity_avg": 0  
  },  
  ...  
],  
[  
  {  
    "dy": "2018-08-21",  
    "hr": 21,  
    "fm": 0,  
    "pt": 0  
  },  
  {  
    "dy": "2018-08-21",  
    "hr": 22,  
    "fm": 0,  
    "pt": 0  
  },  
  {  
    "dy": "2018-08-21",  
    "hr": 23,  
    "fm": 0,  
    "pt": 0  
  },  
  {  
    "dy": "2018-08-22",  
    "hr": 0,  
    "fm": 0,  
    "pt": 0  
  },  
  {  
    "dy": "2018-08-22",  
    "hr": 1,  
    "fm": 0,  
    "pt": 0  
  },  
  ...  
],  
[  
  {  
    "Null": "Null"  
  }  
]  
]
```

Table 9- Sensor Data &amp; Goal Fields

Field	Description
Local Date (dy)	date in timezone where unit is located
Local Hour (hr)	hour
Minutes	minutes of water flow in this goal time period
Flow Minutes (fm)	minutes of water flow in this time period
Pipe Temperature Average C (pt)	average pipe temperature (°C)
Temperature Low C	weather forecast low (°C)
Temperature High C	weather forecast high (°C)
Temperature Average C	weather forecast average (°C)
Humidity Average	weather forecast humidity
Forecast	current weather forecast (sunny, rainy, etc)

### 3.4. I'm Getting Alarm

Let your contacts know if you can get (ie respond to) the current alarm. Updates the server and sends an email to your contacts.

#### 3.4.1. Requirements

Field	Requirements
Device	Osiris, Trident
Access Level	emergency, user, full, customer
Rate Limit	10 requests per hour

#### 3.4.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure":
"Im_Getting_Alarm", "args": "774, Y" }' -H "Content-Type: application/json"
```

Table 10- I'm Getting It Args

Field	Description
Unit Id	unit id
Getting It	Y = getting it, N = can't get it

#### 3.4.3. Response

[]

### 3.5. Mute Alarm

Mute the current alarm for the next n minutes.

#### 3.5.1. Requirements

Field	Requirement
Device	Osiris, Trident
Access Level	user, full, customer
Rate Limit	10 requests per hour

#### 3.5.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email": "api_email@gmail.com", "password": "api_password", "procedure": "Mute_Alarm", "args": "774, 15" }' -H "Content-Type: application/json"
```

*Table 11- Mute Args*

Field	Description
Unit Id	unit id
Minutes	minutes to mute alarm

#### 3.5.3. Response

[]

### 3.6. Cancel Alarm

Cancel the alarm. Updates the server and sends an email to your contacts.

#### 3.6.1. Requirements

Field	Requirements
Device	Osiris, Trident
Access Level	emergency, user, full, customer
Rate Limit	10 requests per hour

### 3.6.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure": "Cancel_Alarm",
"args": "774" }' -H "Content-Type: application/json"
```

*Table 12- Cancel Alarm Args*

Field	Description
Unit Id	unit id

### 3.6.3. Response

[]

## 3.7. Set Water Valve

Set the Trident water valve on / off to enable / shut off the water in the home.

### 3.7.1. Requirements

Field	Requirements
Device	Trident
Access Level	full, customer
Rate Limit	10 requests per hour

### 3.7.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure": "Set_Water_Valve",
"args": "774, on" }' -H "Content-Type: application/json"
```

*Table 10- Set Water Valve Args*

Field	Description
Unit Id	Trident id
Valve	on = enable water flow, off = shut off water flow

### 3.7.3. Response

[]

### 3.8. Set Goal

Set the conservation goal for this month in minutes.

#### 3.8.1. Requirements

Field	Requirement
Device	Osiris
Access Level	user, full, customer
Rate Limit	10 requests per hour

#### 3.8.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure": "Set_Goal", "args":
"774, 10000" }' -H "Content-Type: application/json"
```

*Table 13- Set Goal Args*

Field	Description
Unit Id	unit id
Minutes	minutes below which we meet our monthly goal

#### 3.8.3. Response

[]

### 3.9. Set Unit Settings

Set the settings for this unit.

#### 3.9.1. Requirements

Field	Requirements
Device	Osiris, Trident
Access Level	full, customer
Rate Limit	10 requests per hour



### 3.9.2. Request

```
curl -i --tlsv1 -X POST "https://novamail.biz/php_api/OsirisApi.php" -d '{ "email":
"api_email@gmail.com", "password": "api_password", "procedure": "Set_Config",
"args": "774,
appDisplayTempUnit|devFlowOnPercentOff|devAlarmColdC|devAlarmHotC|devAlarmT
empMin|devAlarmFlowMin, F|132|3|53|25|35" }' -H "Content-Type: application/json"
```

*Table 14- Set Unit Settings Args*

Field	Description
Unit Id	unit id
Fields	delimited list of fields
Values	delimited list of values

### 3.9.3. Response

[]

End of manual