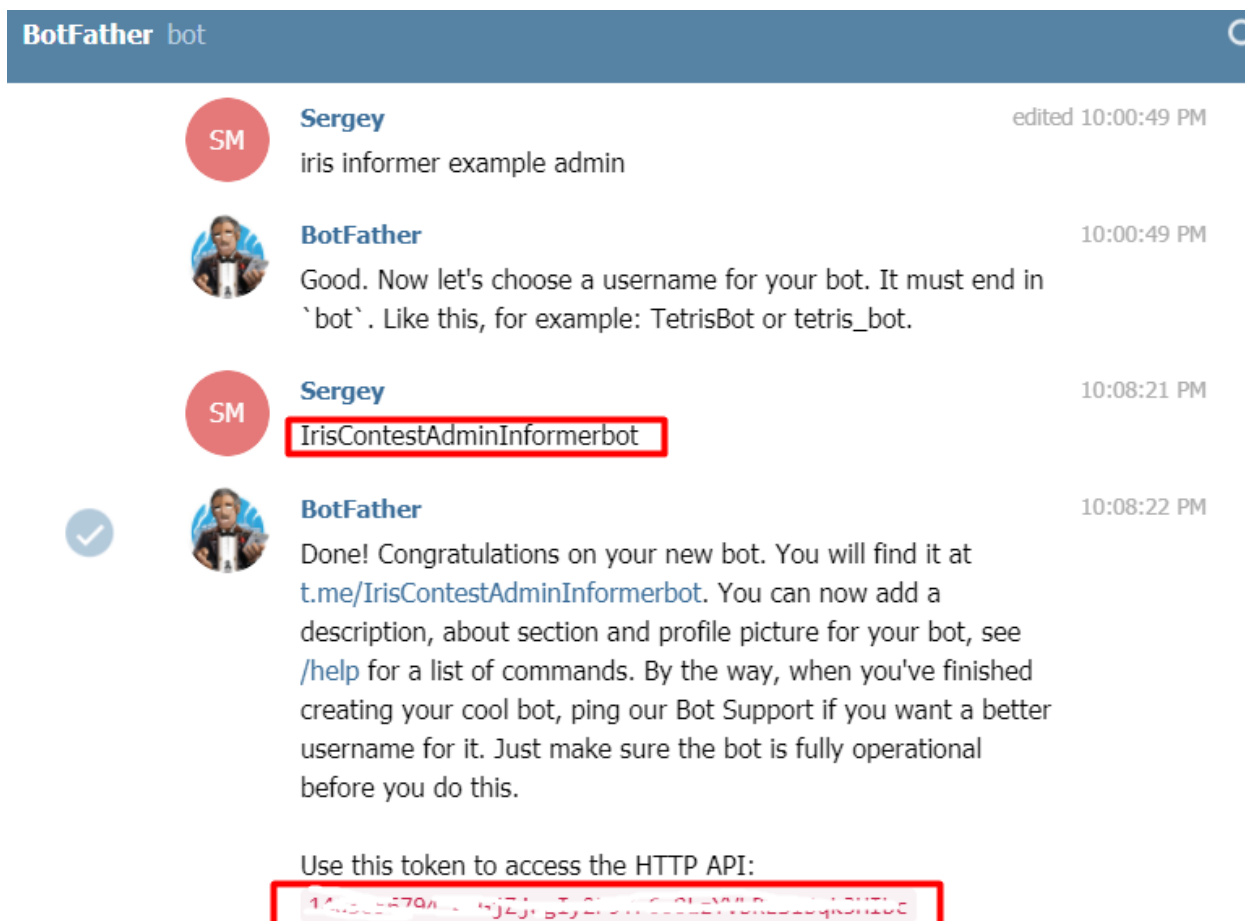


Перед началом установки

Если у вас нет аккаунта в мессенджере Telegram, то это просто повод для его создания. Итак, создадим бота-администратора. Для этого найдите BotsFather, присоединитесь к нему и введите команду /newbot.



Затем мы создадим бота-информатора и сохраним их токены для ввода при установке модуля с помощью диспетчера пакетов.

Вы создаете два бота. Первый админ а второй информер. Первым будете пользоваться только вы как администратор, а имя второго бота можно раздать тем пользователям, которым вы разрешите следить за пользоваться вашим сервисом.

You are creating two bots. The first is the admin and the second is the informer. The first one will be used only by you as an administrator, and the name of the second bot can be given to those persons whom you assign the right to use your service.

Откройте терминал IRIS:

```
USER>
USER>zpm
zpm:USER>install appmsw-telealerts
```

Или если из докера

```
zpm:USER>load /opt/irisapp
```

```
[appmsw-telealerts]      Module object refreshed.  
[appmsw-telealerts]      Validate START  
[appmsw-telealerts]      Validate SUCCESS  
[appmsw-telealerts]      Compile START  
[appmsw-telealerts]      Compile SUCCESS  
[appmsw-telealerts]      Activate START  
[appmsw-telealerts]      Configure START  
Shall we enter tokens ? [y,n] <y> y
```

```
Enter the token to access of the admin bot 123:456zzzzzzzzz  
Enter the number phone of the admin bot 79771234567  
Enter the token to access of the informer bot 123:789xxxxxxxx
```

If you make a mistake, or BotFather changed the token, you can always retry later by performing a utility `##class(appmsw.telestat.API.util).Init()`
Product items changed successfully

```
[appmsw-telealerts]      Configure SUCCESS  
[appmsw-telealerts]      Activate SUCCESS
```

Программа запомнила токены и номер телефона администратора в таблице `appmsw.telestat.Bots`

ID	Access	Site	Chat	Group	Address	Name	Version	Body	Start	Stop	Status	Subpartname	TimeStamp
1													
2													
3													

Для регистрации инстанса IRIS, который будем мониторить, нужно выполнить процедуру:

```
Do ##class(appmsw.telestat.API.inst).EditInst("test2021",1,"_system","SYS","http://iris-test:52773")
```

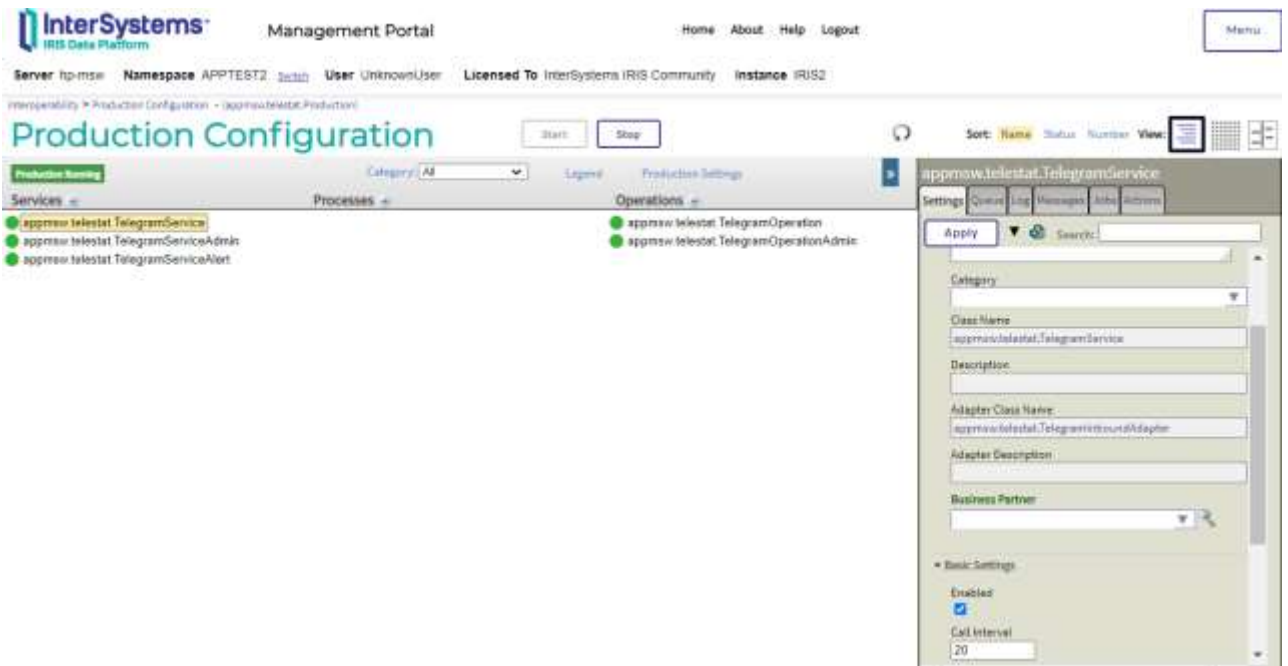
Их может быть несколько.

Если нужно исключить из списка мониторинга, то введем 0 во втором параметре:

```
Do ##class(appmsw.telestat.API.inst).EditInst("test2021",0,"_system","SYS","http://iris-test:52773")
```

На всех инстансах которые включены в список должен быть установлен пакет `apptools-admin`

После этого мы откроем продукцию и запустим её.



Настройка администратора бота

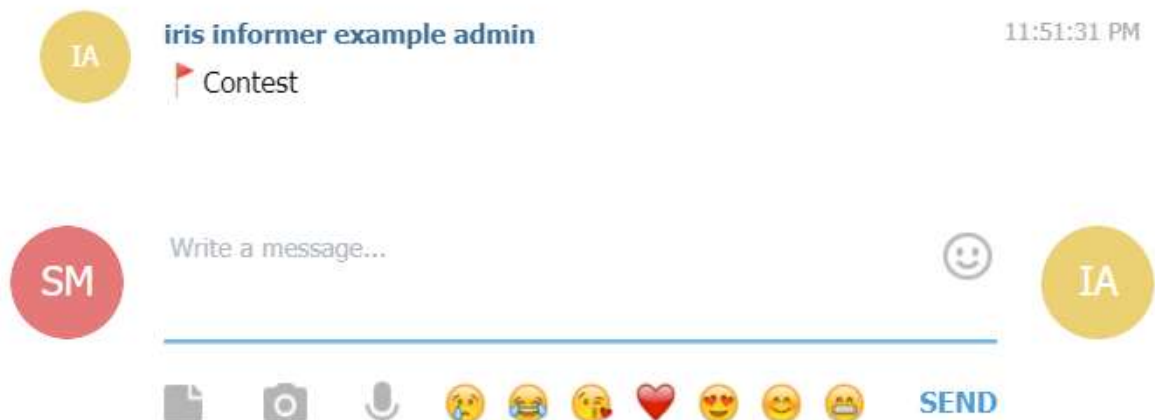
Найдите созданного админ-бота в Telegram, подключитесь к нему и выполните команду /start и мы покажем ему наш номер телефона. Именно тот, который мы ввели при установке. Это нужно сделать именно с мобильного телефона, а из браузерной версии мессенджера номер не передается.

Если все в порядке, мы получим сообщение: Your number has been successfully accepted OK

иначе: Your number is not included in the allowed table. Check the correctness of the initial data

Теперь ChatId привязан к телефону администратора. Вы можете протестировать уведомления админ-боту с помощью команды в терминале

```
user>zwrite ##class(appmsw.telestat.API.util).ToAdmin("Contest")
```



Настройка бота-информатора

Найдем в телеграмме созданного бота-информатора и подключимся к нему, нажав /START кнопку. Сервисная служба подготовит сообщение, а также предложит показать номер телефона.

Администратор бота получит сообщение об отправке телефона, и, нажав кнопку «Разрешить» или «Запретить», вы примете решение, которое будет принято в ответ на сообщение бота-информатора.



Но даже если в доступе к информации было отказано, ChatId и номер телефона были внесены в appmsw.telestat.Bots таблицу, и боту-информатору можно было отправить сообщение с помощью утилиты.

```
user>zwrite ##class(appmsw.telestat.API.util).ToInformer("7971111111",,, "Hello don't be sad")
```

Работа по настройке завершена, и вы можете увидеть, что доступно из коробки.

Для администратора бота, если вы введете /start iris informer example admin

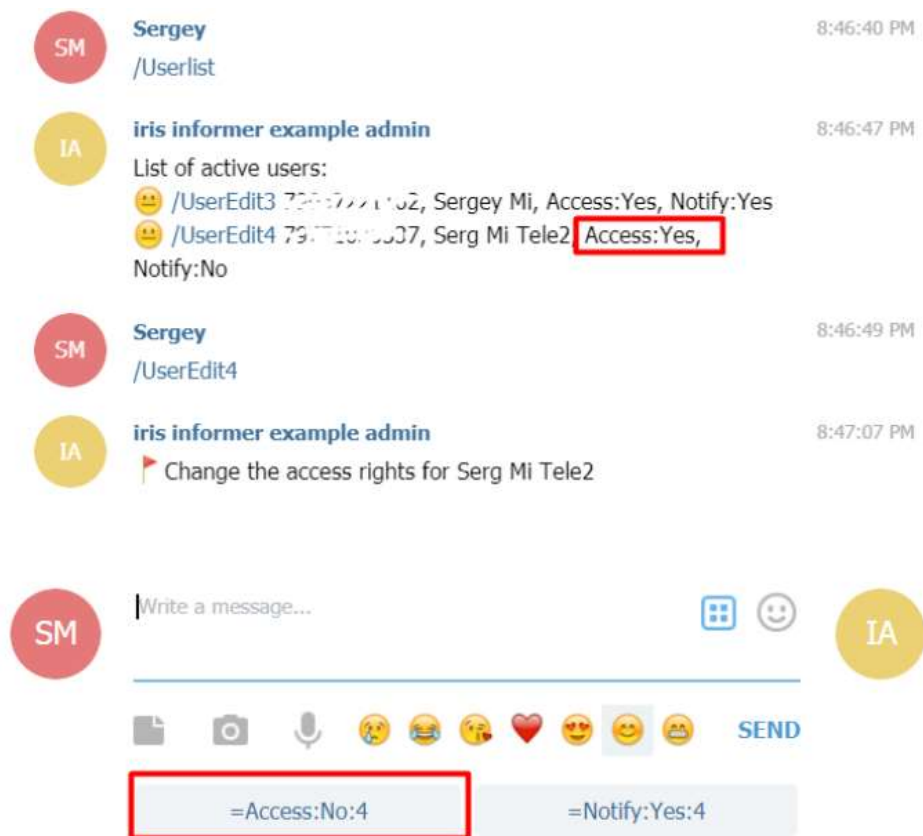
Bot administration service for tracking Ensemble and IRIS servers. Can take commands:

/GetLastAlerts - Get last alerts. Server: 'hp-msw'

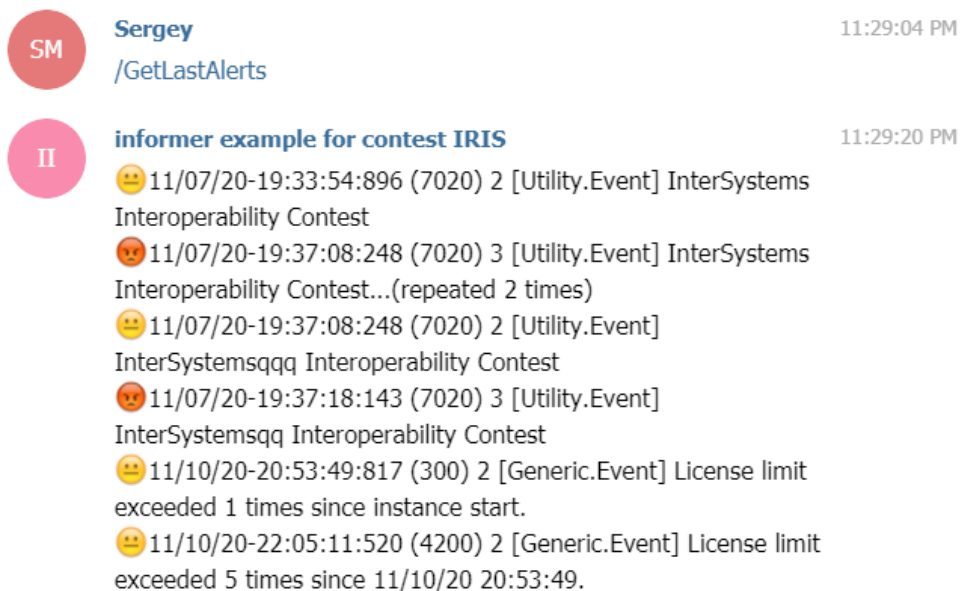
/ServersStatus - Get a list of monitored instances

/Userlist - Get a list of users receiving information about servers and their status

Для бота-администратора есть возможность просматривать и редактировать атрибуты пользователя с помощью /Userlist команды



Другая команда /GetLastAlerts выведет из каждого инстанса последние 10 сообщений из протокола messages.log.



И appmsw.telestat.TelegramServiceAlert служба настроена на периодическую проверку системных сообщений и, если их уровень значимости более чем 2, отображать их для всех пользователей, подключенных к боту - информатору, которые имеют право получать уведомления.

Проверим это выполнив терминале метод: `do ##class(%SYS.System).WriteToConsoleLog("InterSystems Iris Test",,3)`

Список команд и содержимого расширяется. Достаточно создать свой собственный класс, аналогичный `appmsw.telestat.API.commands` методом `GetCommands` И для списка команд `GetAlerts` Для получения по ним содержимого.

Команды и контент можно различать между пользователями по группам. Это решение было настроено и протестировано в `Long polling` режиме. Но его можно настроить и в `Webhook` режиме. Конфигурация `SSL` создается автоматически.