**Этот комплекс состоит из следующих компонентов:**

***- Zabbix-сервер***

***- Zabbix-агент***

***- Zabbix-прокси***

**Zabbix-сервер**

В минимальной конфигурации достаточно только Zabbix-сервера. Он умеет не только пинговать узлы, то также опрашивать их по snmp, http, telnet/ssh, ipmi протоколам. Позволяет проводить проверки по самописным скриптам (например опрос времени жизни ssl-сертификатов). Конфигурируется он через удобный (хотя с первого раза непонятный) веб-интерфейс.

**Zabbix-агент**

Для мониторинга серверов обычно ставят на них zabbix-агента. Zabbix-агент также чрезвычайно гибкий инструмент по сбору параметров с наблюдаемого компьютера. Часть базовых проверок встроено в агента, но если их не хватает, то с помощью скриптов можно лихо увеличить количество собираемой информации.

Но это еще не все. При правильной настройке zabbix-агент умеет запускать скрипты по команде с сервера. А сервер может выслать такую команду по сработавшему событию (к примеру, когда основной интернет отключился, перейти на резервный.)

Агент существует как под windows, так и под \*nix.

**Zabbix-прокси**

Zabbix-прокси пригодится, если у нас в подчинении много офисов, которые на связи бывают не всегда. Чтобы не делать много самостоятельных серверов и не путаться во вкладках браузера мы можем собирать показания с помощью zabbix-proxy, который при появлении связи передаст их на центральный сервер, на котором мы и будем анализировать информацию. Также прокси может помочь для оптимизации нагрузки на zabbix-сервер.

Сегодня мы постараемся замониторить состояние наших серверов с samba и asterisk из предыдущих уроков. Используется связка Zabbix-сервер + zabbix-агент. В этот раз наблюдаемые узлы будем добавлять вручную. Автодобавление будет рассмотрено в одном из углубленных курсов.

Параметры за которыми будем наблюдать:

1. Количество запущенных процессов сервиса (если ноль, то выдавать предупреждение)

2. Слушают ли сервисы необходимые tcp/udp порты (если нет, то выдать предупреждение)

3. Доступность серверов по icmp.

Это позволит нам познакомиться со стандартными шаблонами и написать пару своих. В том числе со скриптами расширяющими возможности zabbix-агента.

Для начала установим Zabbix-сервер и подготовим его к работе (официальная инструкция тут:

**https://www.zabbix.com/documentation/6.0/ru/manual/installation/install\_from\_packages/debian\_ubuntu**

В репозитории нашего дистрибутива есть пакеты zabbix, но они неактуальных версий. К счастью разработчики заббикса поддерживают свой репозиторий. **У нас традиционно дистрибутив Ubuntu 22.04**. Базу данных выберем mysql.

Чтобы подключить репозиторий надо выполнить вот эти команды:

wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-release/zabbix-release\_6.0-3+ubuntu22.04\_all.deb

dpkg -i zabbix-release\_6.0-3+ubuntu22.04\_all.deb

sudo apt-get update

Ставим все необходимые пакеты:

sudo apt install zabbix-server-mysql zabbix-frontend-php zabbix-nginx-conf zabbix-sql-scripts zabbix-agent

**Установите** и запустите сервер базы данных.

Инструкция:

<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-22-04>

sudo apt update

sudo apt install mysql-server

sudo systemctl start mysql.service

Создаем БД:

sudo mysql -uroot -p

**password**

**mysql>** create database zabbix character set utf8mb4 collate utf8mb4\_bin;

**mysql>** create user zabbix@localhost identified by 'password';

**mysql>** grant all privileges on zabbix.\* to zabbix@localhost;

**mysql>** set global log\_bin\_trust\_function\_creators = 1;

**mysql>** exit;

На хосте Zabbix сервера импортируйте начальную схему и данные. Вам будет предложено ввести недавно созданный пароль.

zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix

Отключение опции log\_bin\_trust\_function\_creators после импорта схемы базы данных.

sudo mysql -uroot -p

**password**  
mysql> set global log\_bin\_trust\_function\_creators = 0;  
mysql> exit;

Отредактируйте файл **/etc/zabbix/zabbix\_server.conf**

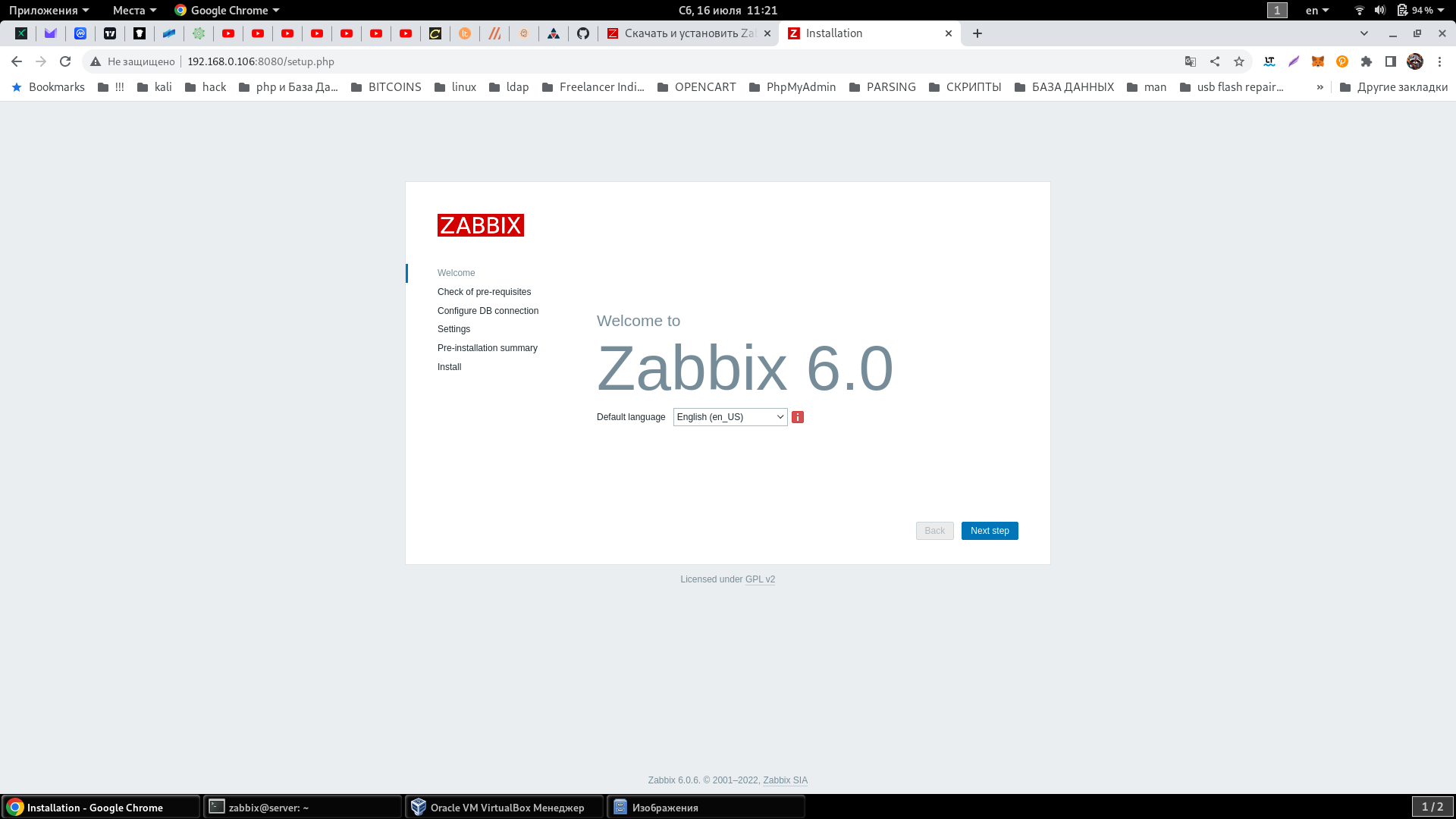
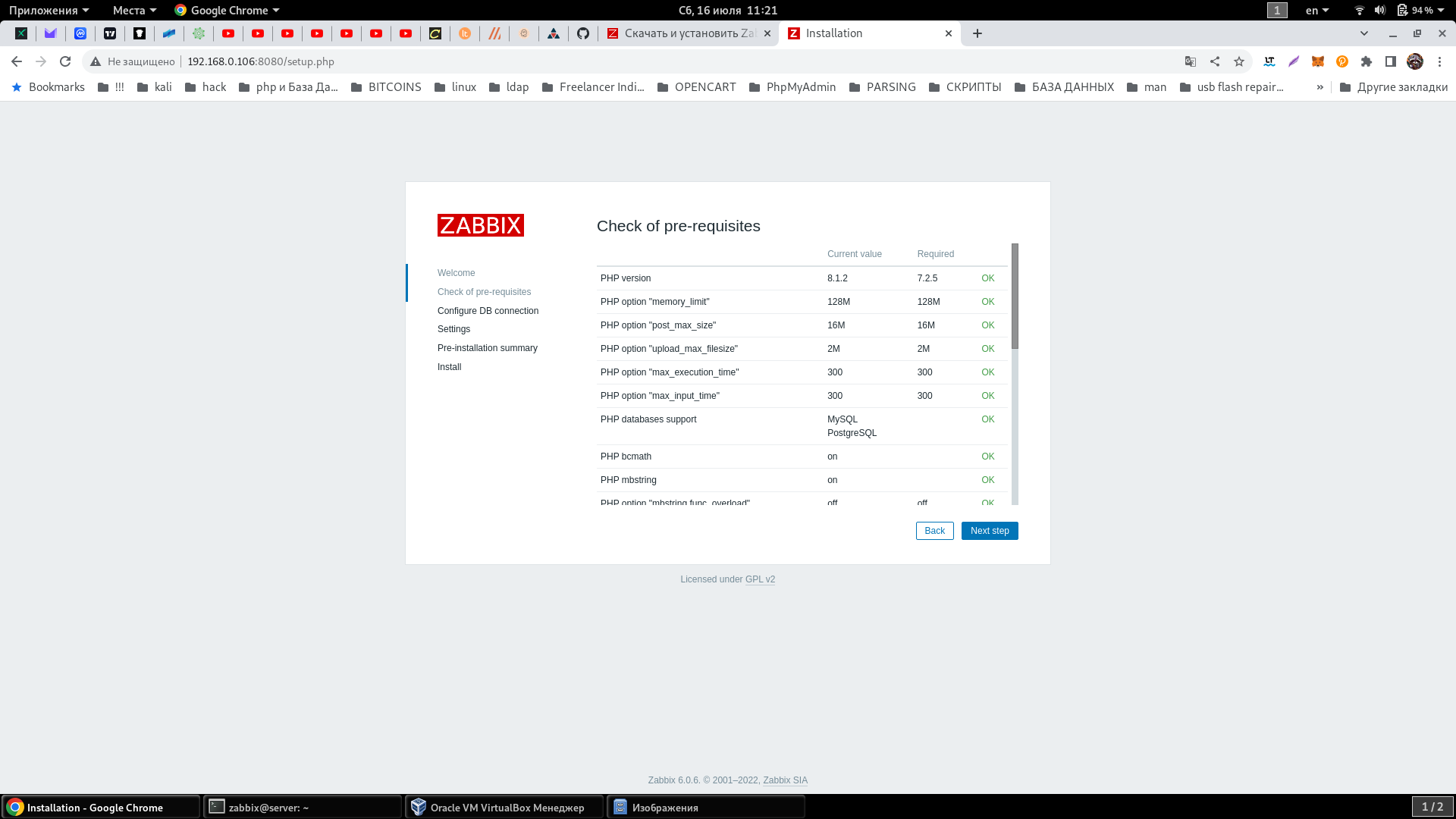
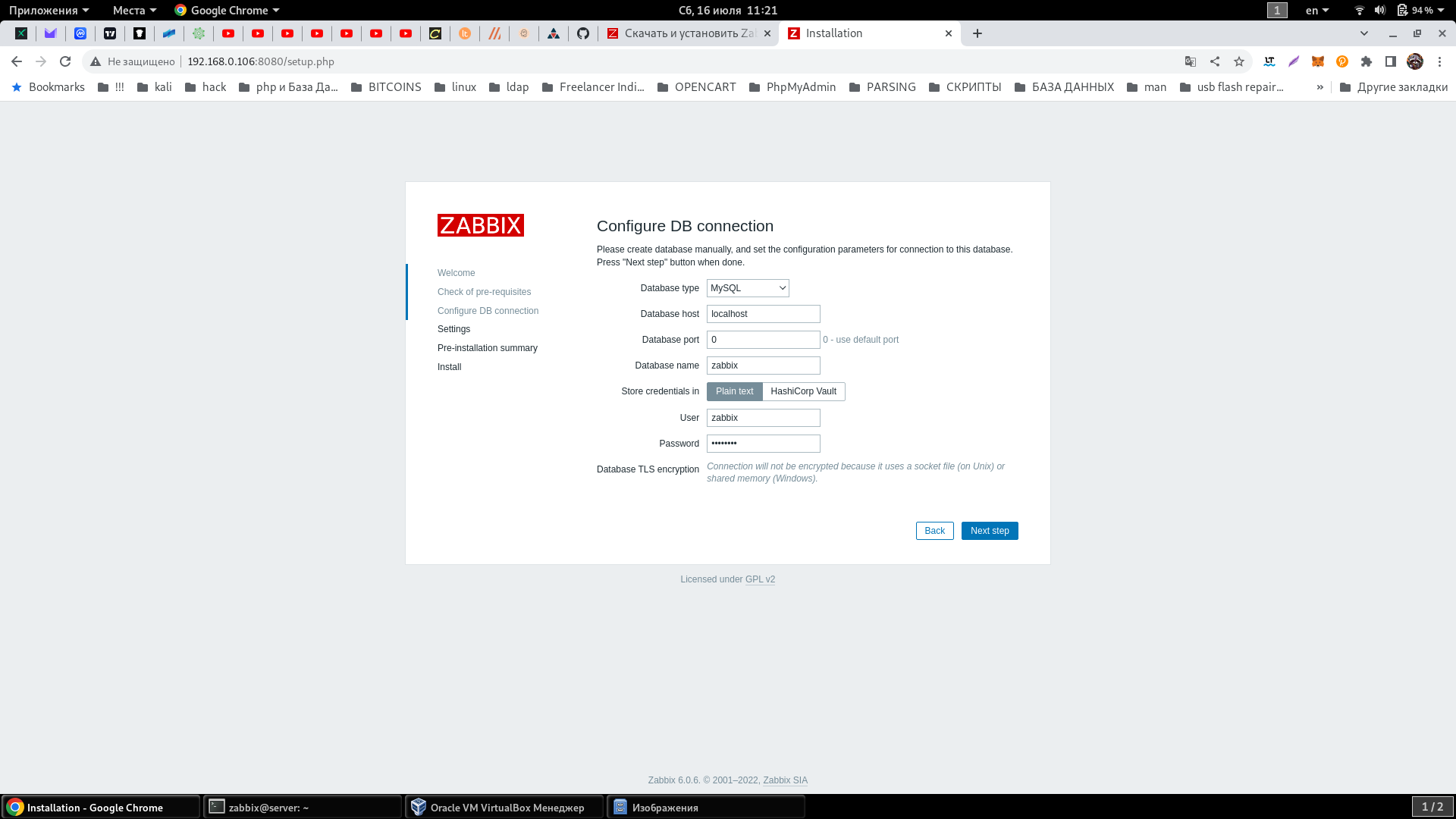
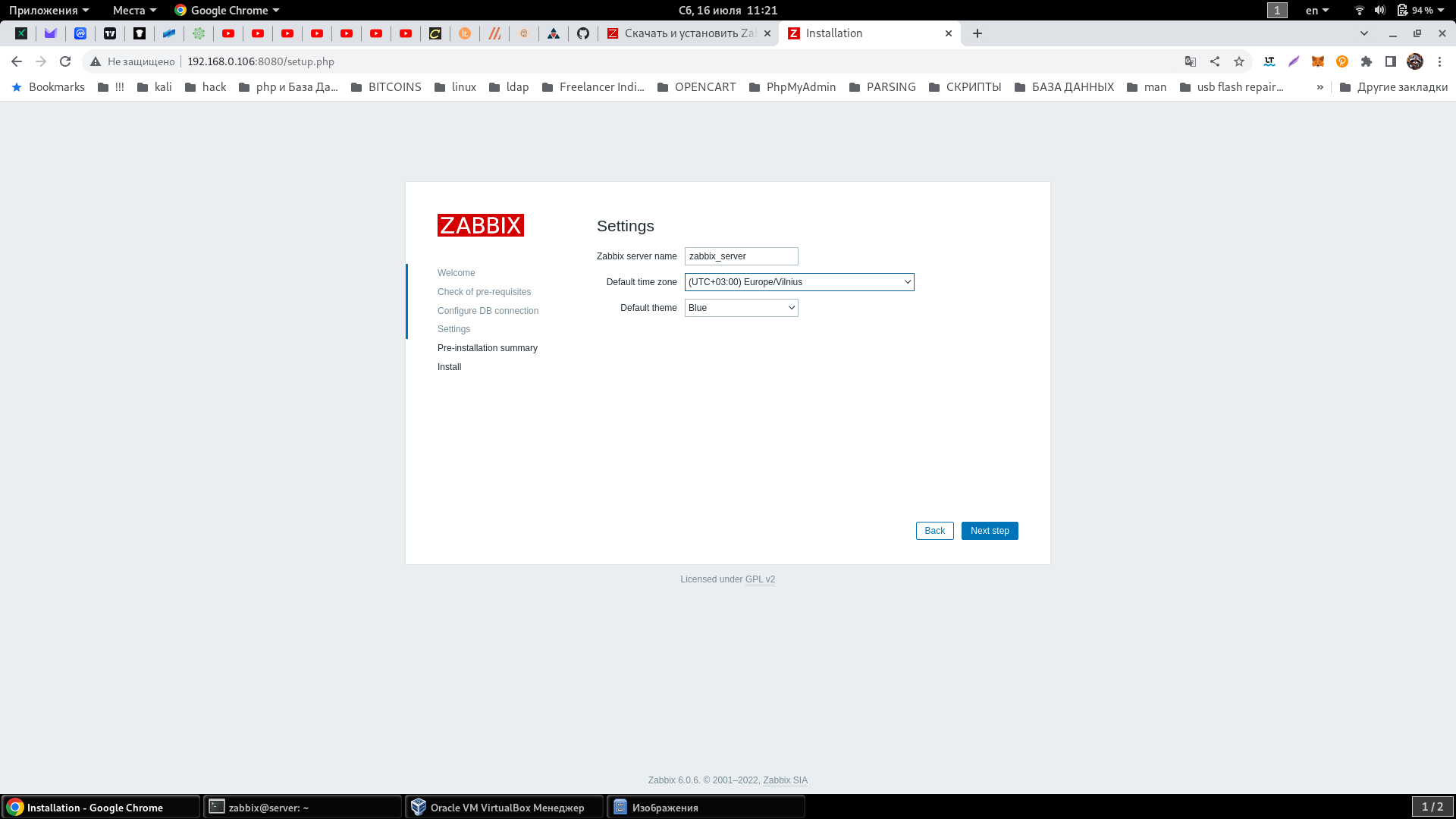
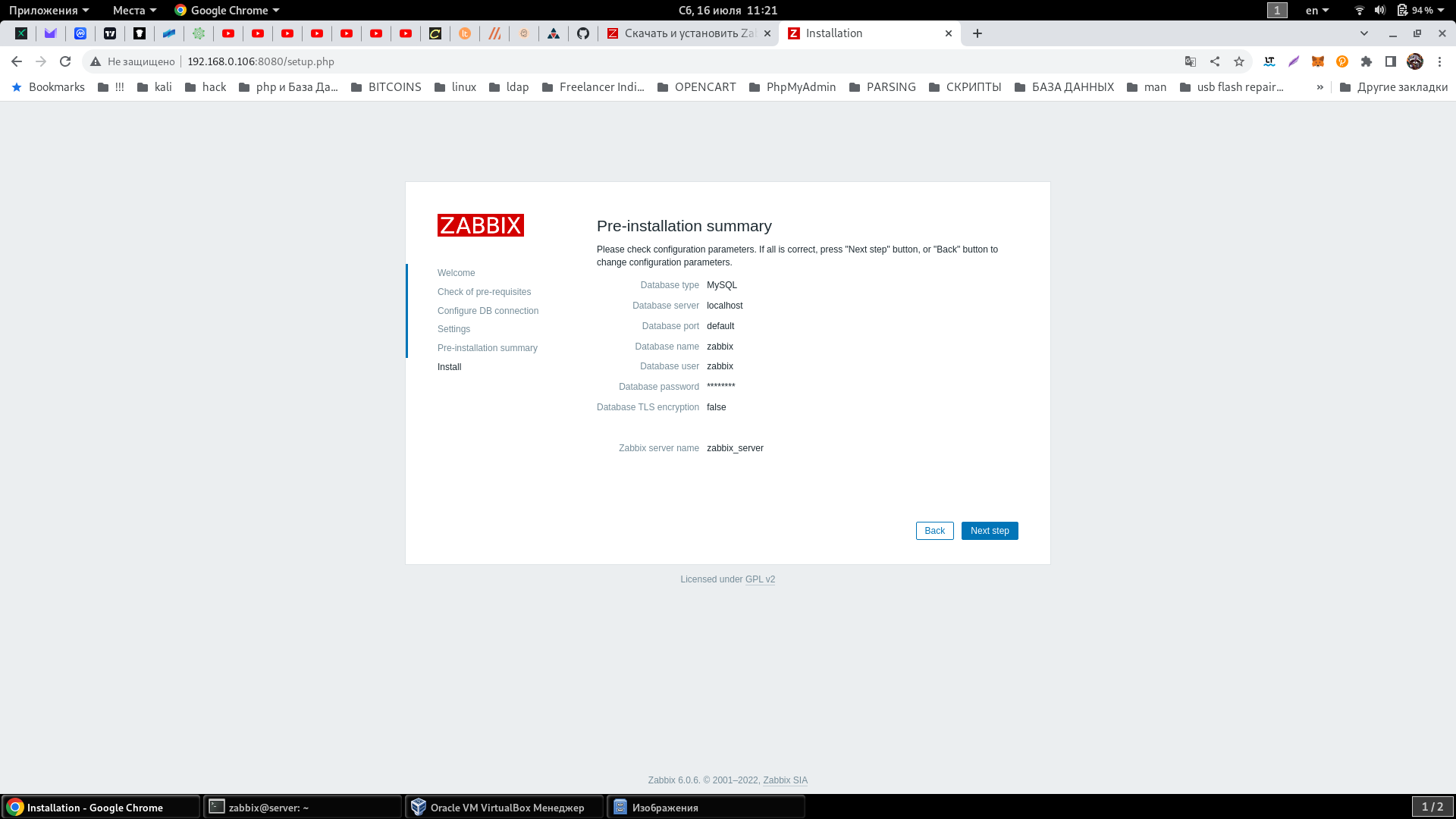
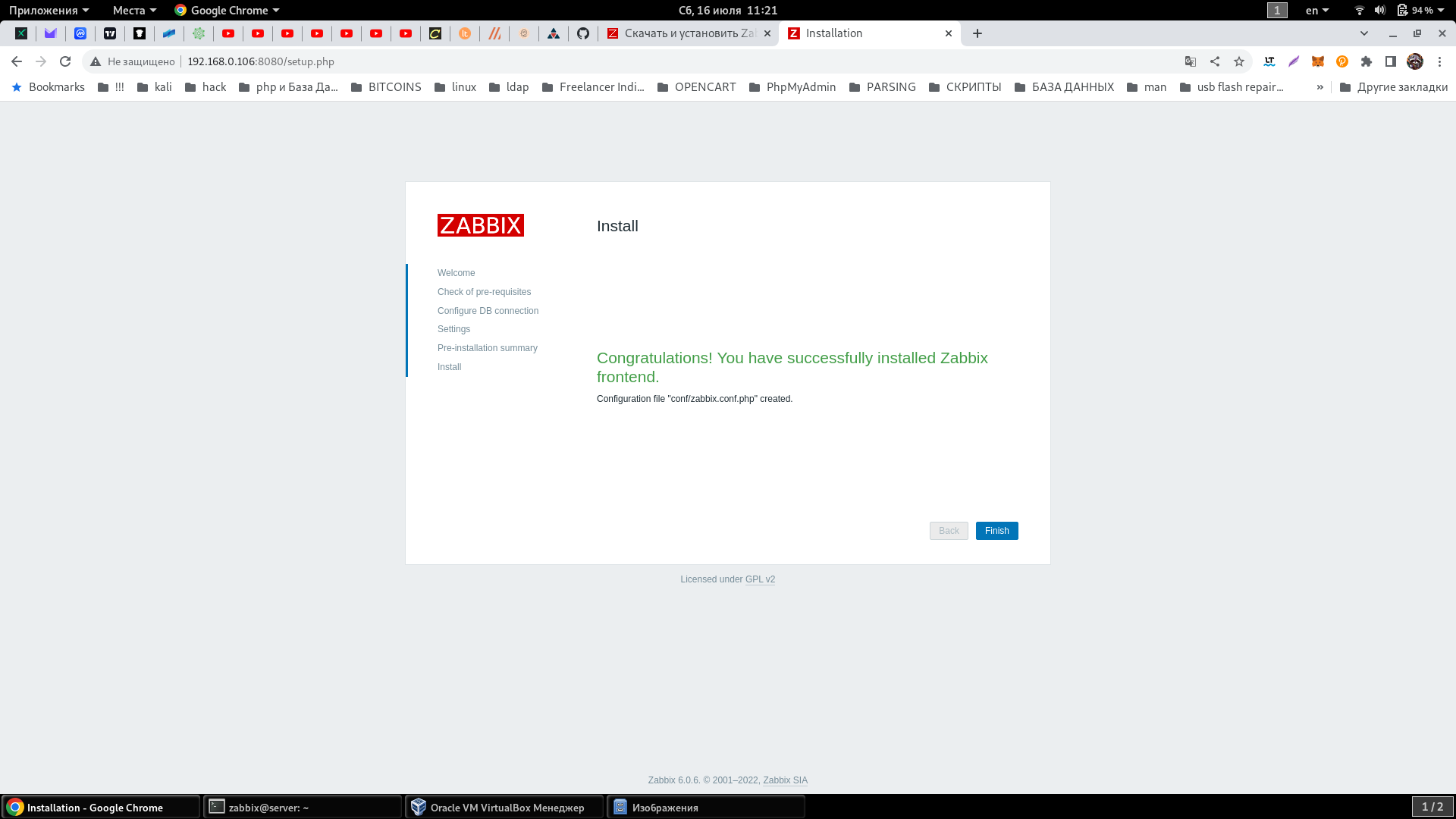
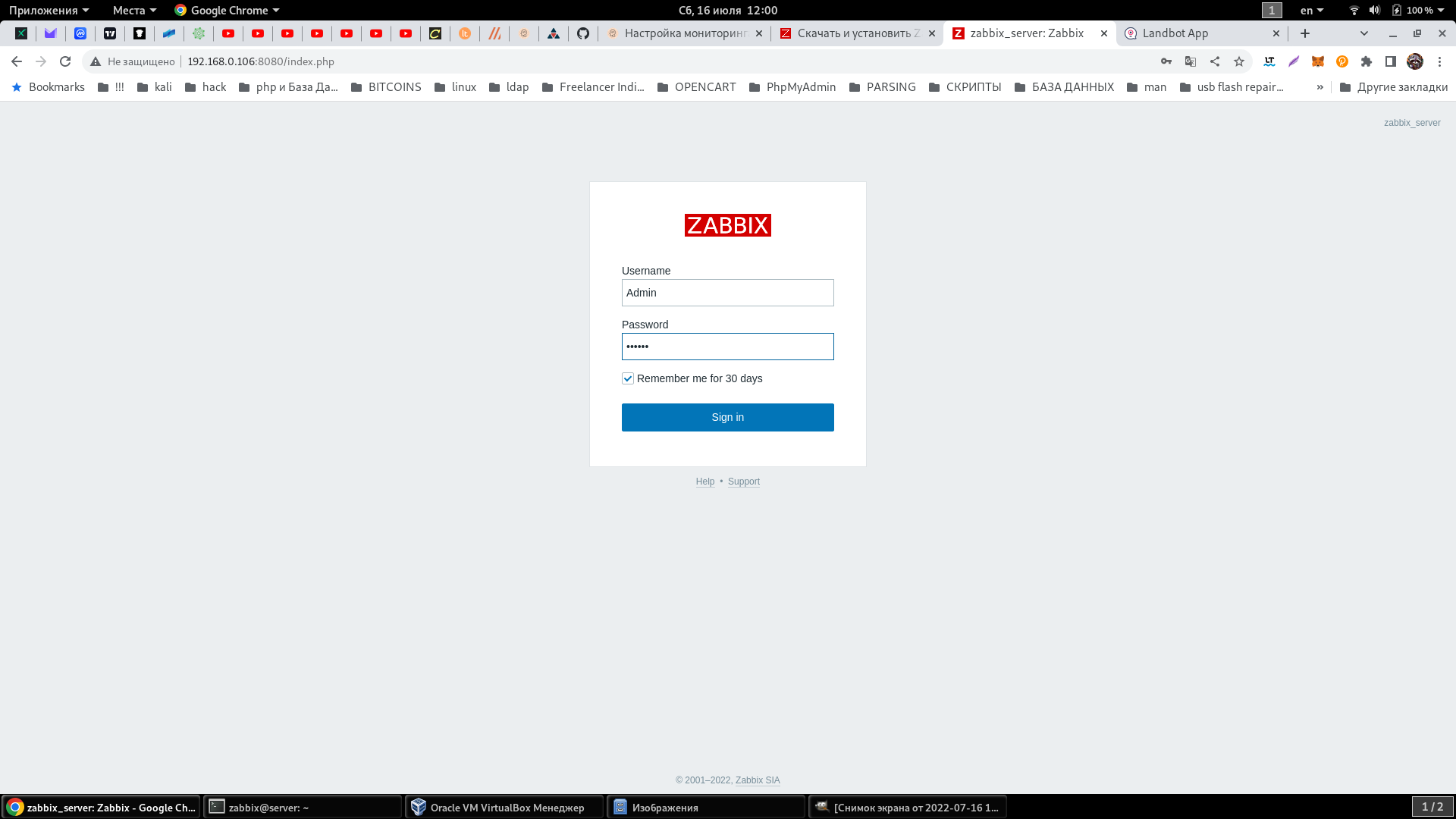
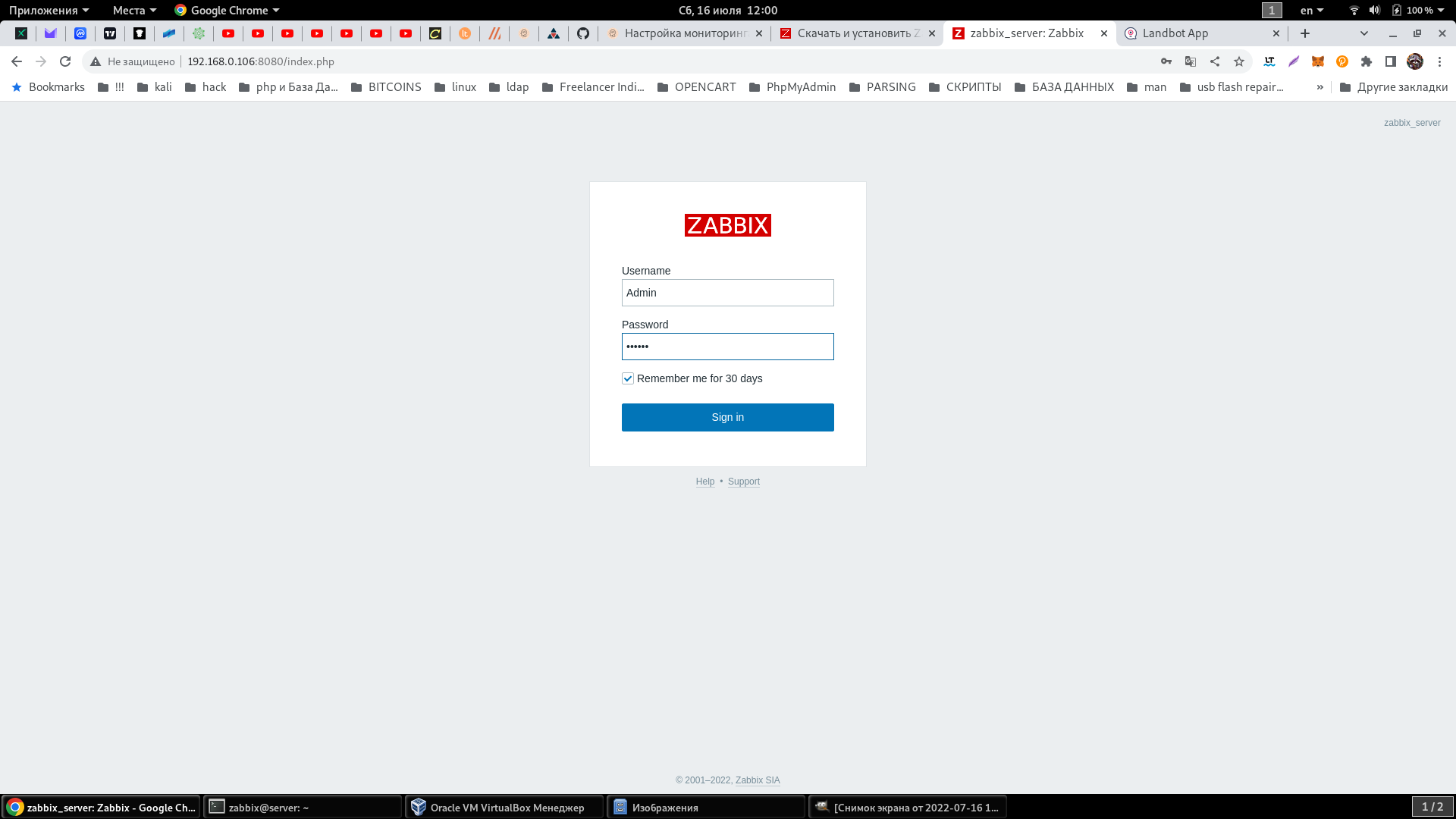
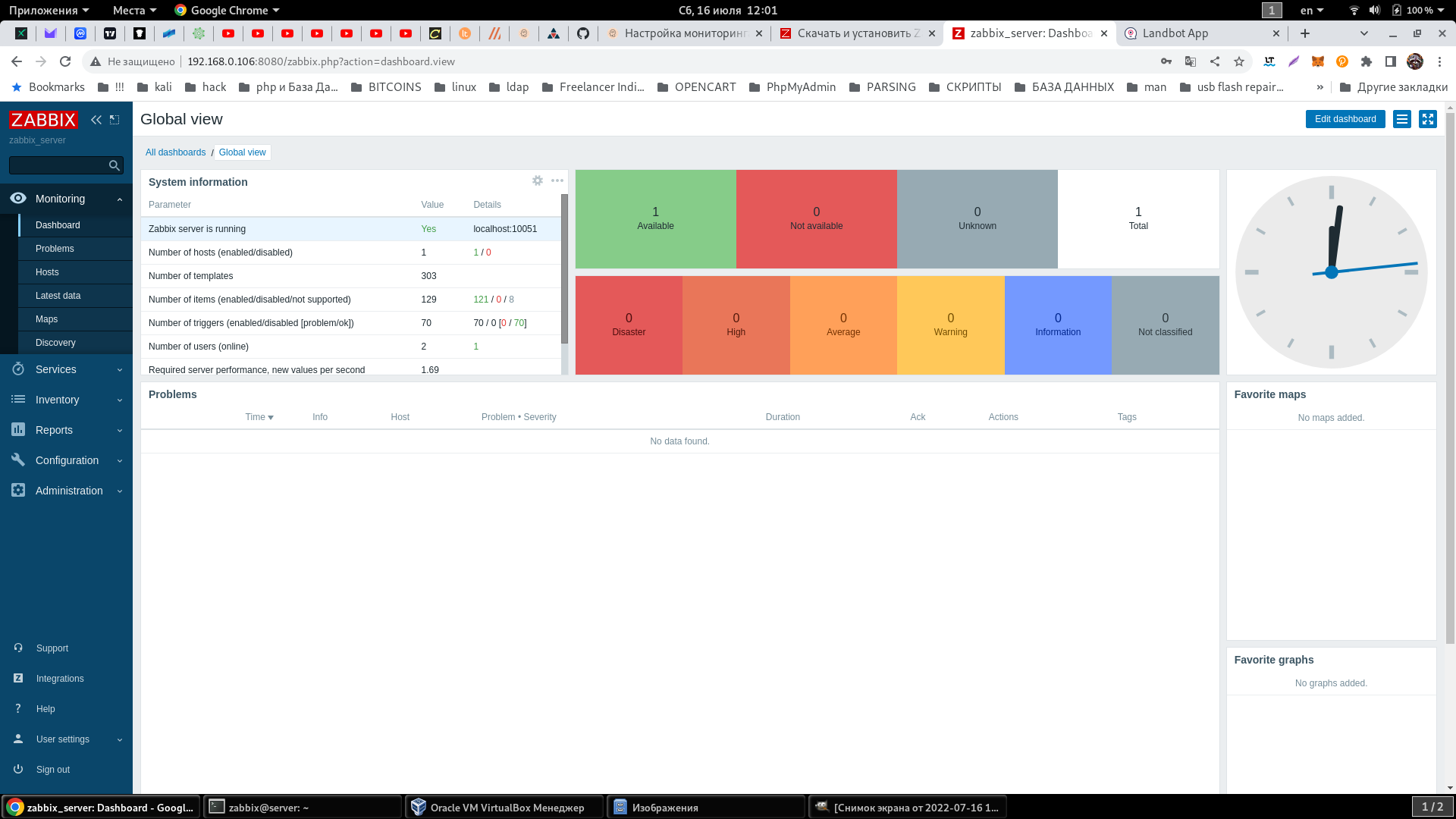
Раскоментируйте строку и впишите свой пароль, если создавали.

DBPassword=password

**Настройте PHP для веб-интерфейса.**

Отредактируйте файл **/etc/zabbix/nginx.conf**, раскомментируйте и настройте директивы 'listen' и 'server\_name'.

# listen 8080;  
# server\_name example.com;

* listen 8080 - это порт по которому будет открываться zabbix.
* server\_name example.com - IP вашего сервера.
* Все пакеты установились, база данных настроена, так что если хочешь продолжить, то надо рестартнуть сервисы.
* systemctl restart zabbix-server zabbix-agent nginx php8.1-fpm
* systemctl enable zabbix-server zabbix-agent nginx php8.1-fpm
* Теперь можешь открыть веб-интерфейс нашего будущего мегакомбайна по мониторингу. Он доступен вот по такой ссылке: http://ip\_нашего\_сервера:8080
* Интересно
* 
* 
* Это экран проверки настроек php. Если ты все делал на чистой установке Ubuntu 22.04, то ошибок быть не должно. Полистай список в поисках красного цвета.
* 
* Настроика подключение к БД.
* Вводим те данные, которые создавали при создании БД.
* Database name = zabbix
* User = zabbix
* Password = password
* 
* Ввводим имя сервера и Time Zone.
* В поле Name пишем название нашего сервера. Выбрать можно любое, какое захочется.
* 
* Проверка параметров
* 
* Поздравляю, вы успешно установили zabbix.
* Жмем finish и переходим
* 
* Это окно авторизации. Учетная запись по умолчанию вот такая:
* **Username**: Admin
* **Password**: zabbix
* 
* 
* Обращаем внимание на Zabbix server is running:
* Должно быть Yes

Для дальнейшей работы надо еще немного скучной теории.

Познакомимся с основными терминами zabbix

**Хост** - наблюдаемый узел

**Элемент данных** - это одиночный (дискретный) элемент данных, собираемый с хоста. К примеру загрузка процессора, сколько занято оперативки или ip-адрес сетевой карты. У элемента данных есть параметр - период опроса. К примеру, если период равен 60 секунд, то после создания элемента данных, мы увидим ответ от хоста через минуту. Соответственно и триггеры по нему сработают только через минуту. Надо это понимать.

Возникает желание поставить этот период в 1 секунду, но тогда нагрузка на базу данных будет огромной. А хост может не успевать отдавать эти данные. Поэтому выбор периода опроса это компромисс между временем реакции и производительностью.

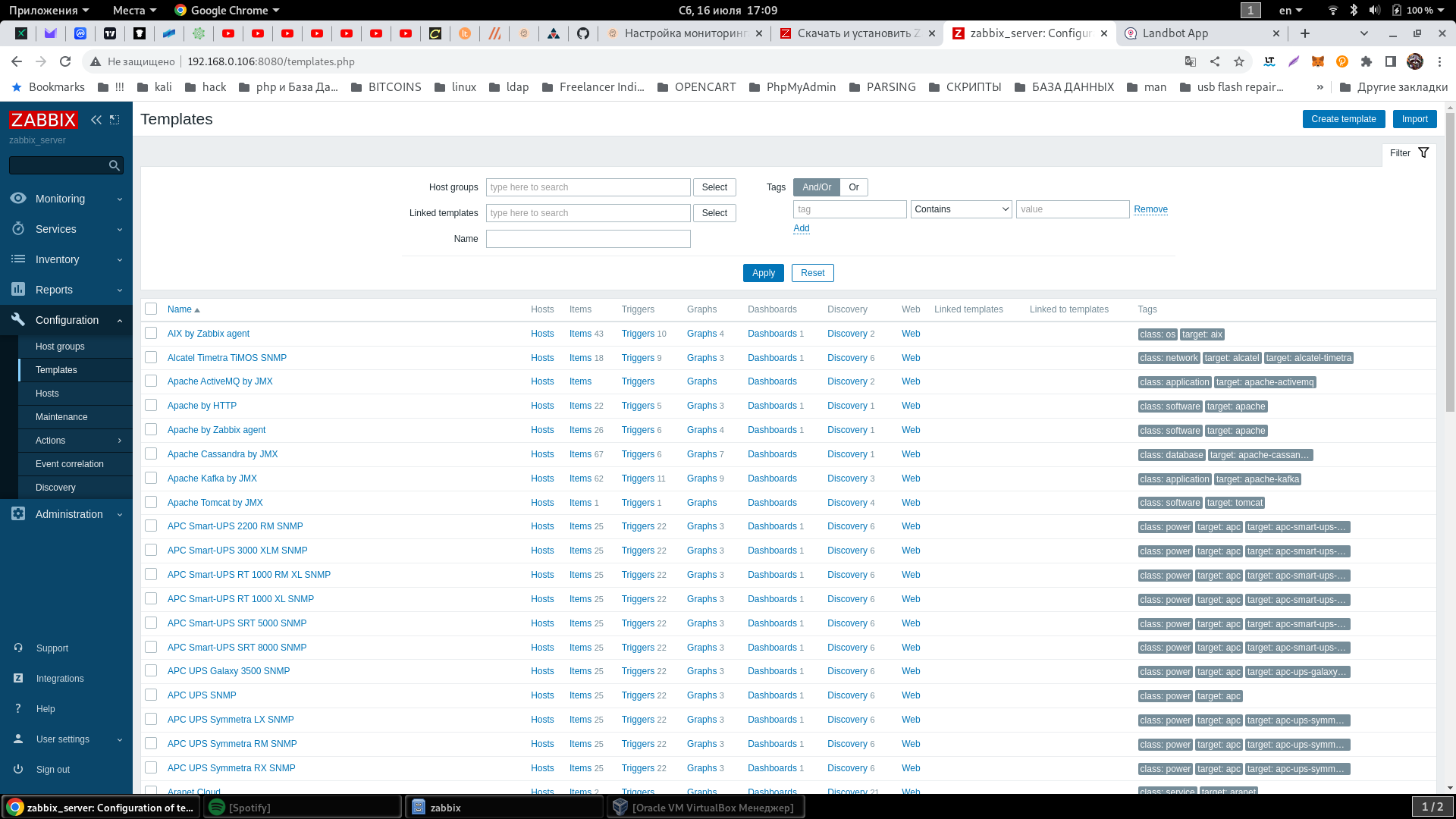
**Триггер** - условие, по которому генерируется событие. Зависит от элементов данных. Не может без них существовать.

**Действие** - действие (отправить уведомление, выполнить скрипт и т.д.), которое происходит после срабатывания триггера. Зависит от триггера. Не может без него существовать.

Можно на каждый хост отдельно создавать свои элементы данных и триггеры, но гораздо удобнее определить общие параметры и накатывать их автоматически на группы хостов. Именно для этого существуют шаблоны. Шаблоны содержат сразу много элементов данных и триггеров по ним.

С заббикс-сервером комплектом идет несколько шаблонов по умолчанию. Сейчас мы с ними познакомимся. Теория закончилась. Начинаем действовать.

Заходим в Configuration - Templates и видим вот такой вот большой список шаблонов.



Можно их пооткрывать и посмотреть из чего они состоят.

Когда надоест можно будет приступать к следующему шагу.

Сейчас будем добавлять наши хосты со стандартными шаблонами. Львиную долю данных можно получить ими. А что останется, то создадим руками.

Уточним задачу:

Добавлять будем два хоста - файлсервер и астериск.

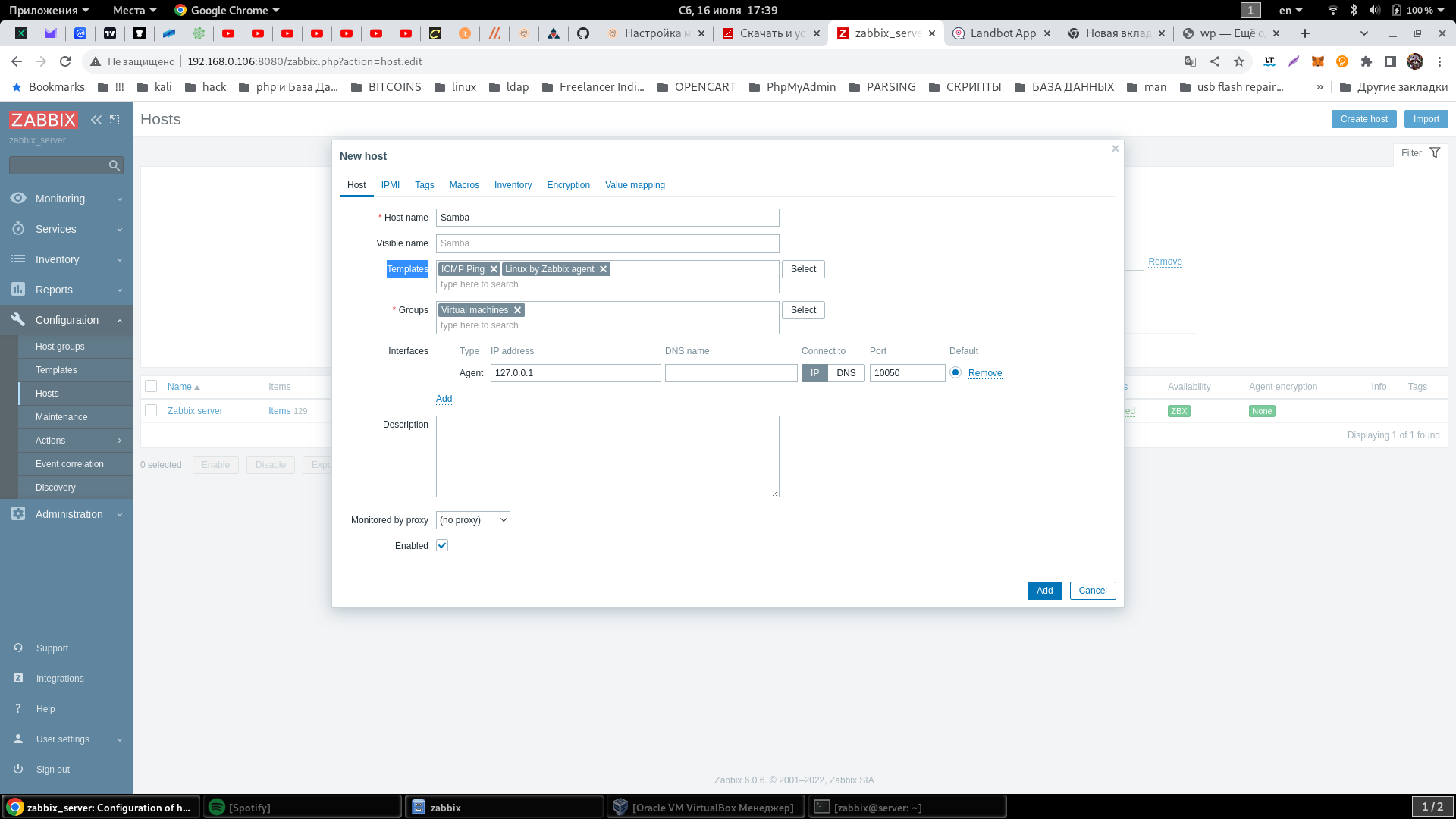
ip адрес файлсервера - 172.22.3.140

ip адрес астериска - 172.22.3.103

Заходим в Configuration - Hosts и жмем Create Host

Полей тут много. И не одна вкладка к тому же =). Будем их потихоньку заполнять.

Вкладка Host



Name - Samba

Templates - ICMP Ping, Linux by Zabbix Agent

Group - жмем на кнопку Select и выбираем там Virtual Machines

Agent Interfaces - IP-адрес - 172.22.3.140

Шаблоны можно добавлять, нажав кнопку Select. Но это не очень продуктивно. Для лентяев тут есть лазейка ;).

Если ввести в поле Link new templates часть названия интересующего нас шаблона, то откроется выпадающий список шаблонов, чьи имена совпадают с набранным.

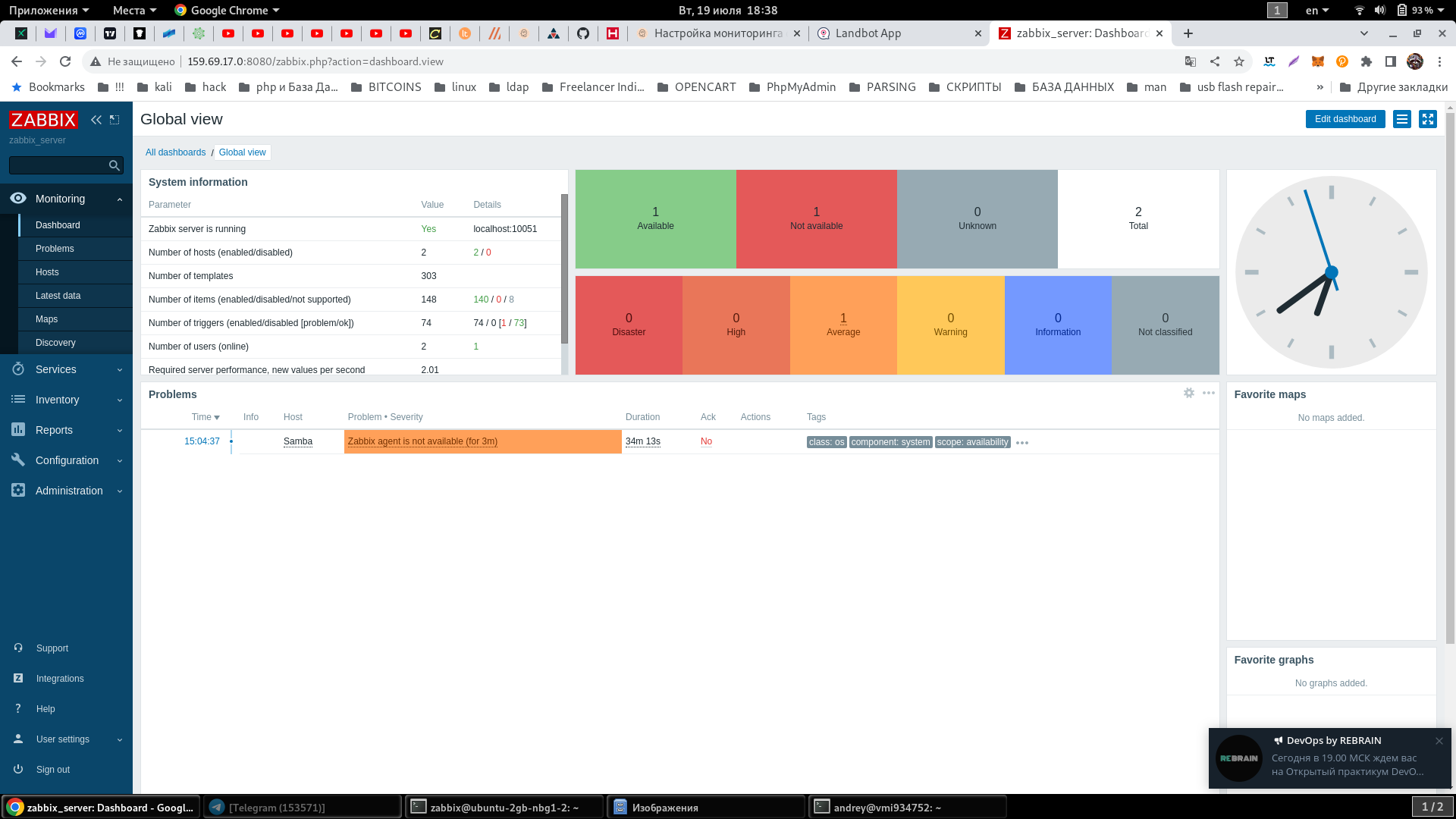
К примеру если ввести icmp, то откроется Template ICMP Ping. А если ввести Linux то откроется много всего =). Среди которых искомый Template OS Linux

Когда набьем шаблонов, можно нажать Add и хост добавится и встанет на мониторинг.

Теперь можно минут на 5 отойти, чтобы все данные по шаблонам подтянулись и мы смогли увидеть сработавшие триггеры.

Чтобы выйти на экран мониторинга надо нажать Monitoring - Dashboard

А почему у нас проблема на хосте Samba?



Все очень просто. У нас же не стоит Zabbix-агент на наших виртуалках с самбой и астериском.

Заходим на виртуалки и добавляем репозиторий заббикса также как и в начале урока.

Только устанавливать мы будем поменьше пакетов =).

sudo apt-get install zabbix-agent

Открываем файл **/etc/zabbix/zabbix\_agentd.conf**

Тут нас интересует два параметра:

- Hostname. Можно найти и раскомментировать его. А можно просто в начале файла написать

Hostname = Samba

Этот параметр должен быть уникальным у всех узлов. Сервер будет по нему различать хосты.

- Server. Тут мы указываем сервер, который имеет право опрашивать агента. У меня сервер живет на ip - 172.22.3.155 так что я пропишу

Server = 172.22.3.155

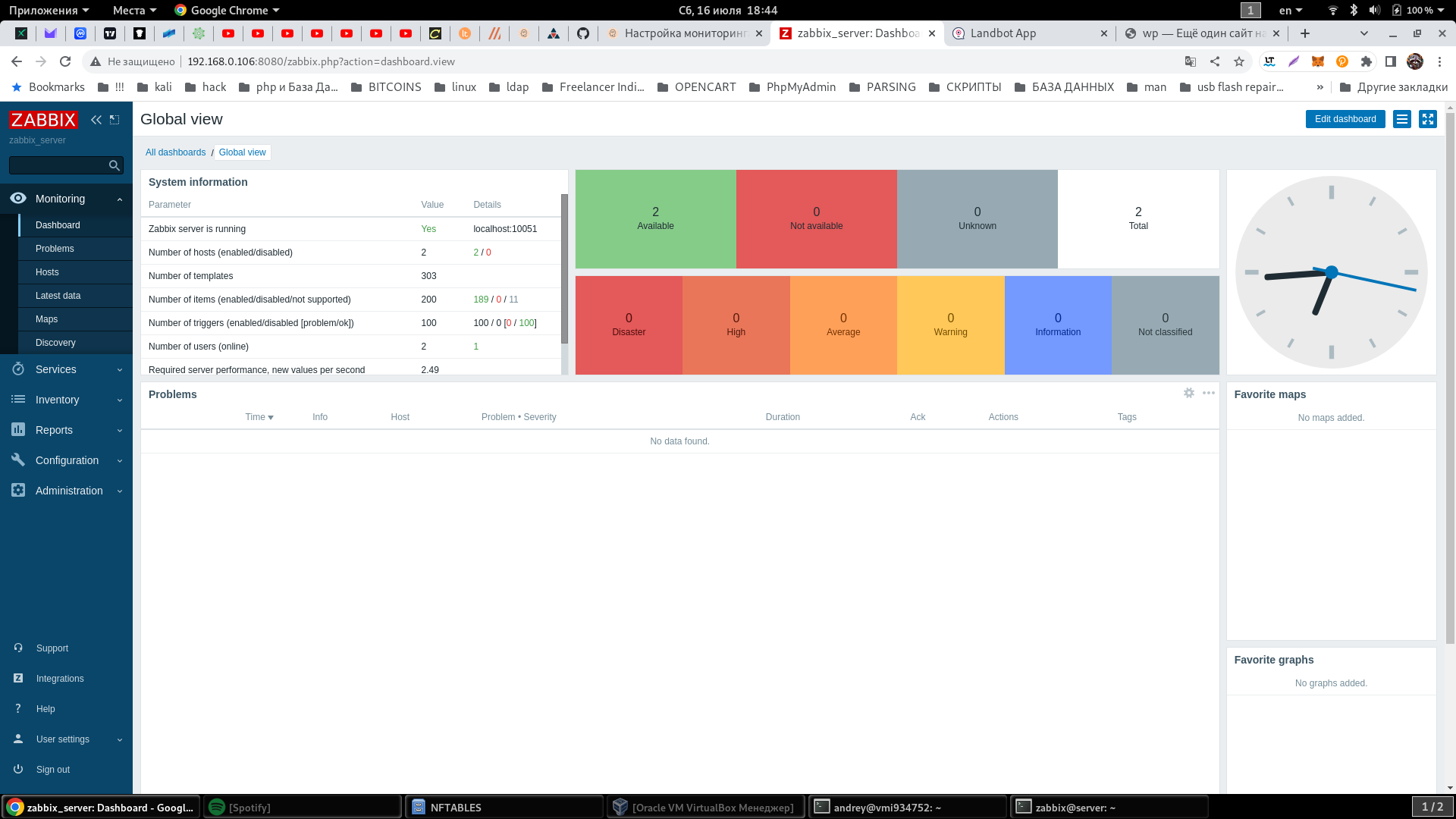
Тебе нужно вписать ip-адрес своего сервера.

Теперь мы запускаем агента и добавляем в автозагрузку

sudo service zabbix-agent restart

sudo systemctl enable zabbix-agent

И можно сходить попить чаю минут на 5. Через 5 минут уведомление в Dashboard пропадет. А в квадратике Available циферка увеличится на один.



Оформи хост Астериск также, как мы сейчас вместе оформили хост Samba.

В целом у нас уже есть система мониторинга. Шаблон Template\_OS\_Linux содержит в себе триггеры на недоступность, на высокую загрузку процессора или оперативки, а также просигнализирует, когда закончится место на диске.

Можно было бы на этом остановиться. Но мы продолжим =)

Уведомления в панели то у нас есть. А вот по почте никто никого не уведомит. Обидно.

Бум исправлять =).

Заходим в **Configuration - Actions - Trigger actions** и нажимаем **Create Action**.

И заполняем форму =).

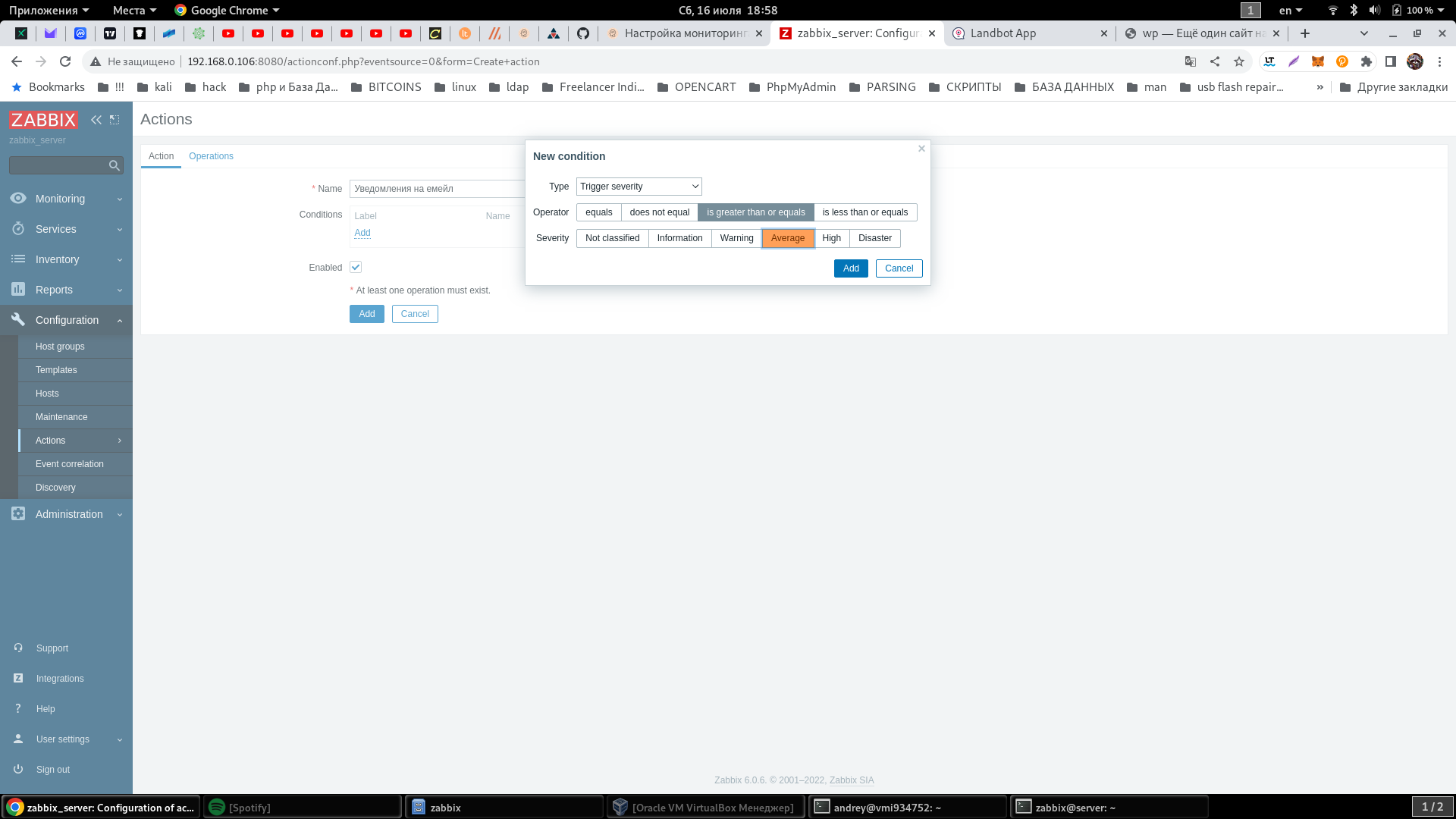
**Name** - название. Пусть у нас называется "Уведомления на емейл"

Ниже блок, в котором добаляются условия, по которым будет совершено действие. Нам нужно уведомление по критичным для нас событиям.

Поэтому выбираем из выпадающего списка значение **Trigger Severity**. Набор полей немного перестроится. В следующем по списку поле выбираем **is greater than or equals.** И в последнем поле выбираем **Average**

Только что мы задали условие по которому это действие будет выполняться для любого триггера важность которого выше чем просто предупреждение.

Для добавления условия нужно нажать **Add** в блоке **New Condition**, рядом с выражением которое мы только что натыкали



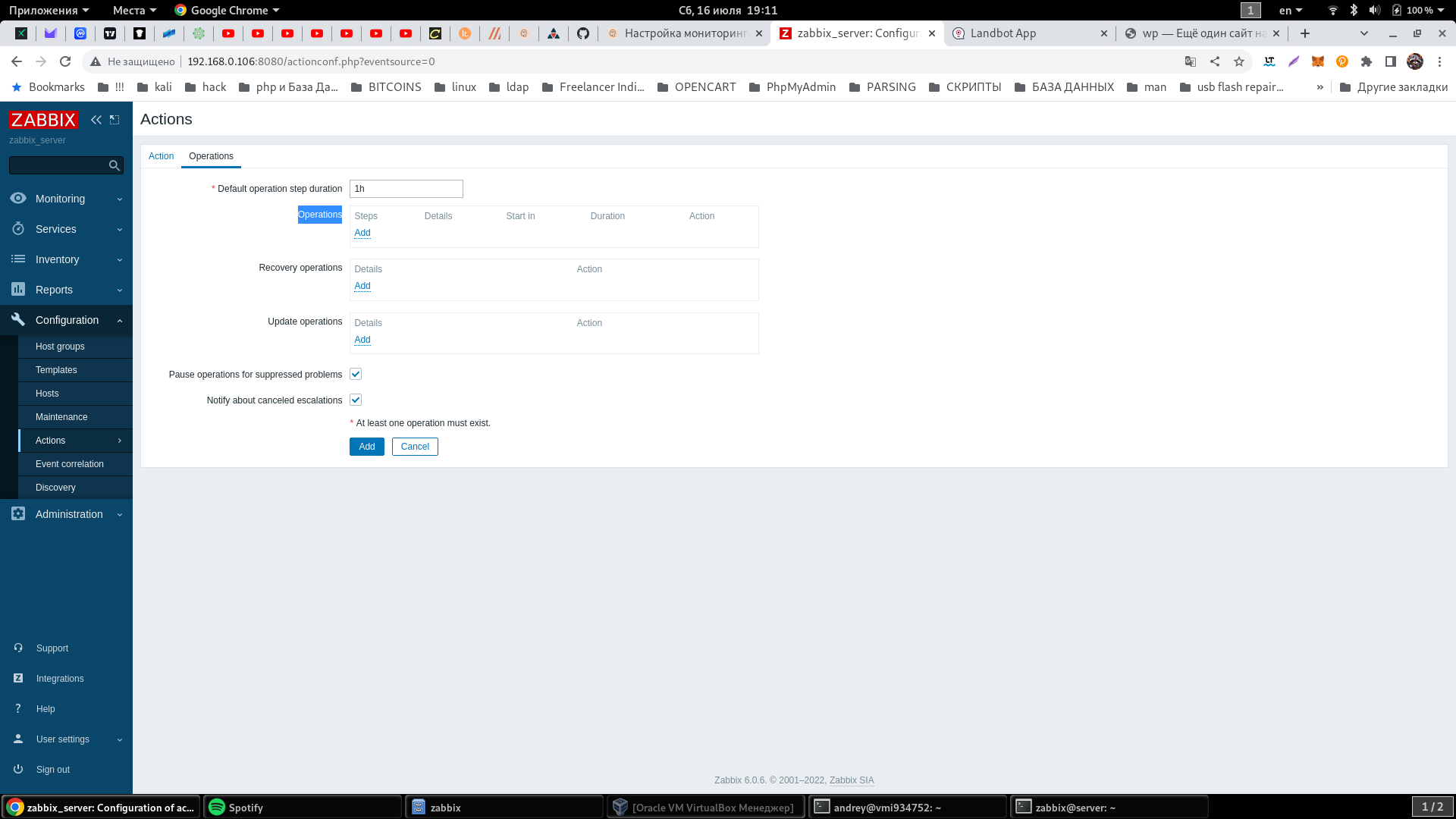
Это мы были во вкладке **Action**. Мы задали условие, но условие это еще не все.

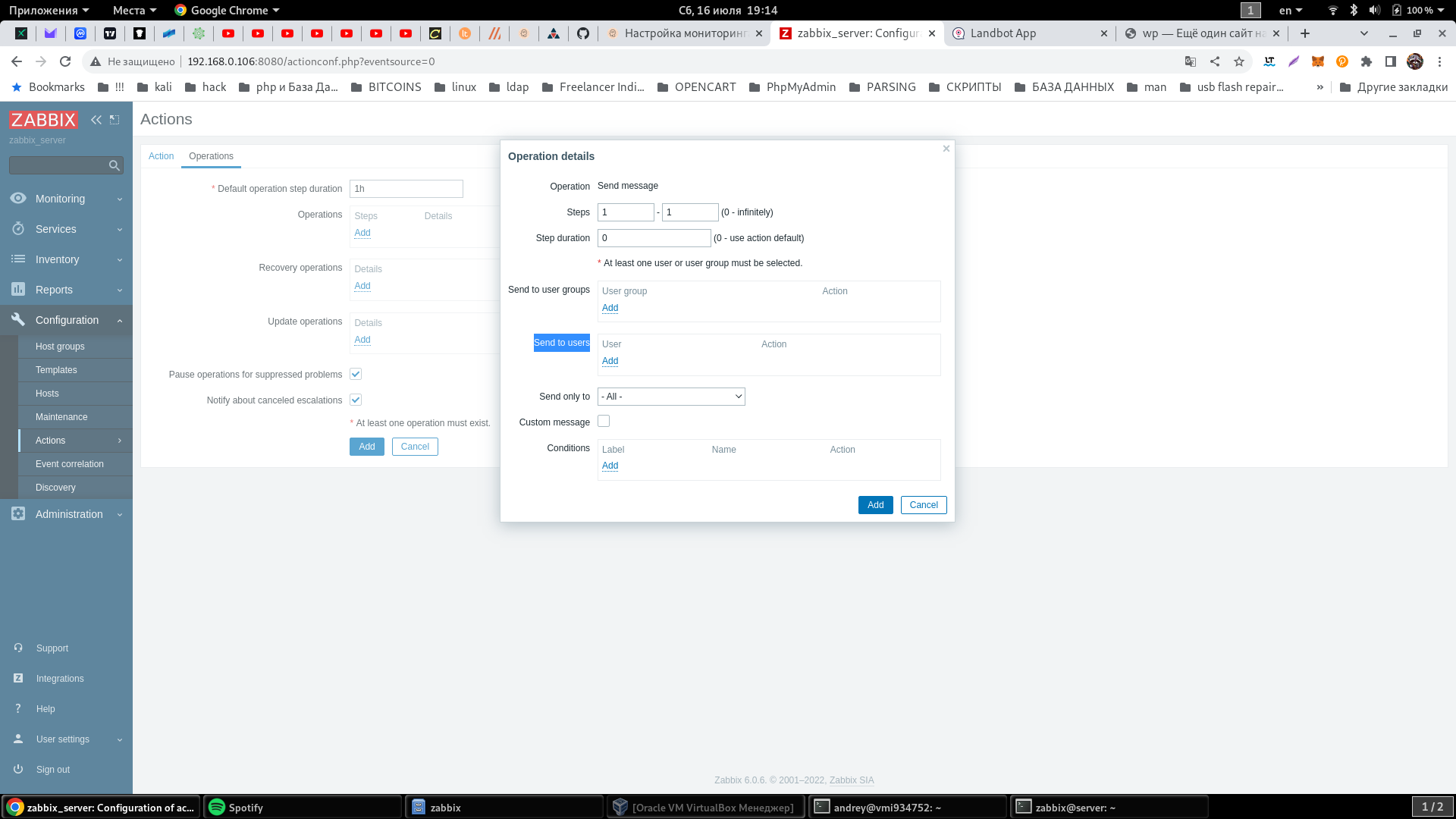
Теперь пропишем что именно надо будет сделать по этому условию. Для этого перейдем на вкладку **Operations.**

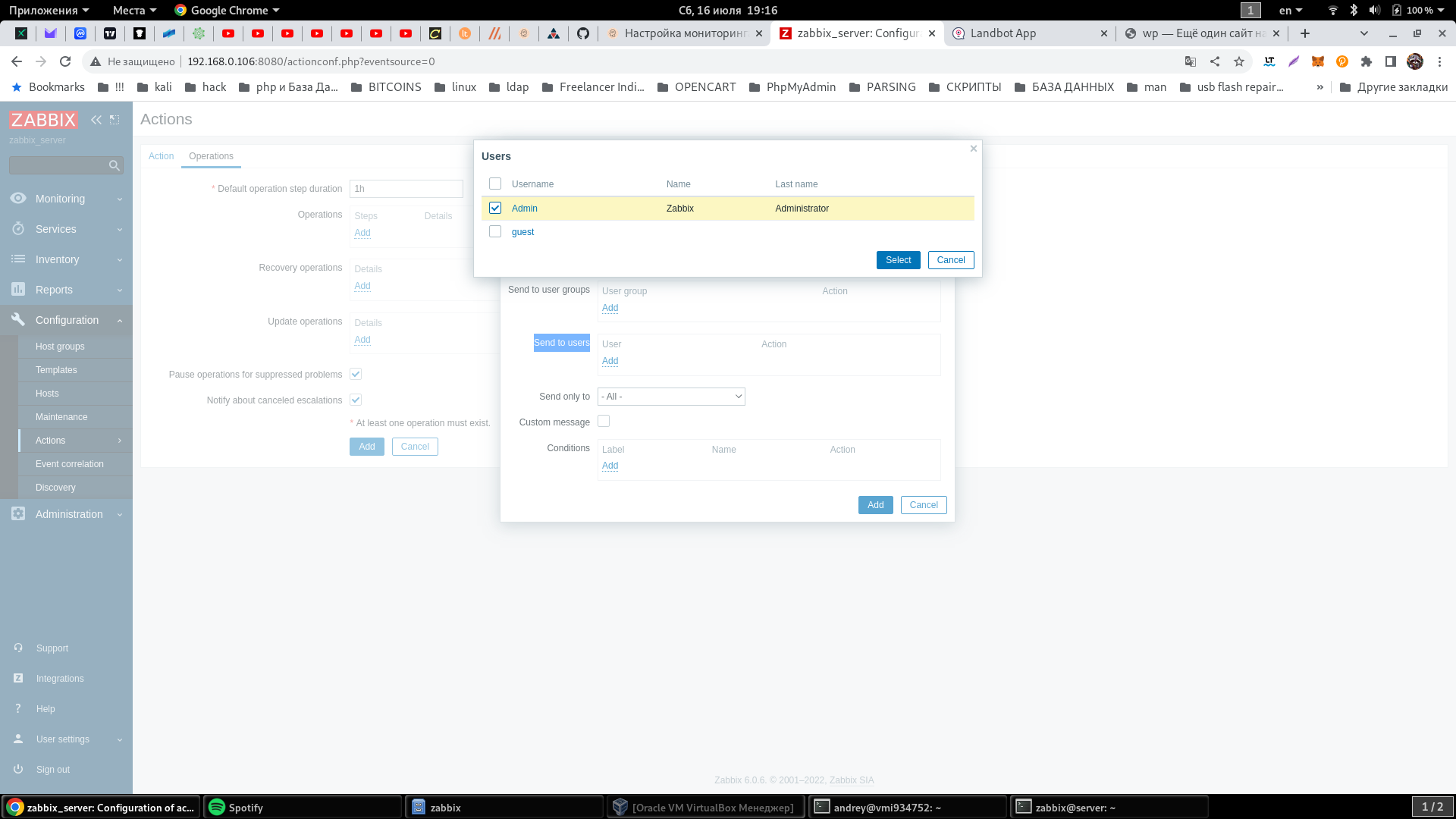
Нас интересует тут блок **Operations**. В нем нужно нажать **Add** и там заполнить всего один блок - **Send to Users**. В нем надо через кнопку **Add** добавить нашего единственного пользователя - **Admin**.

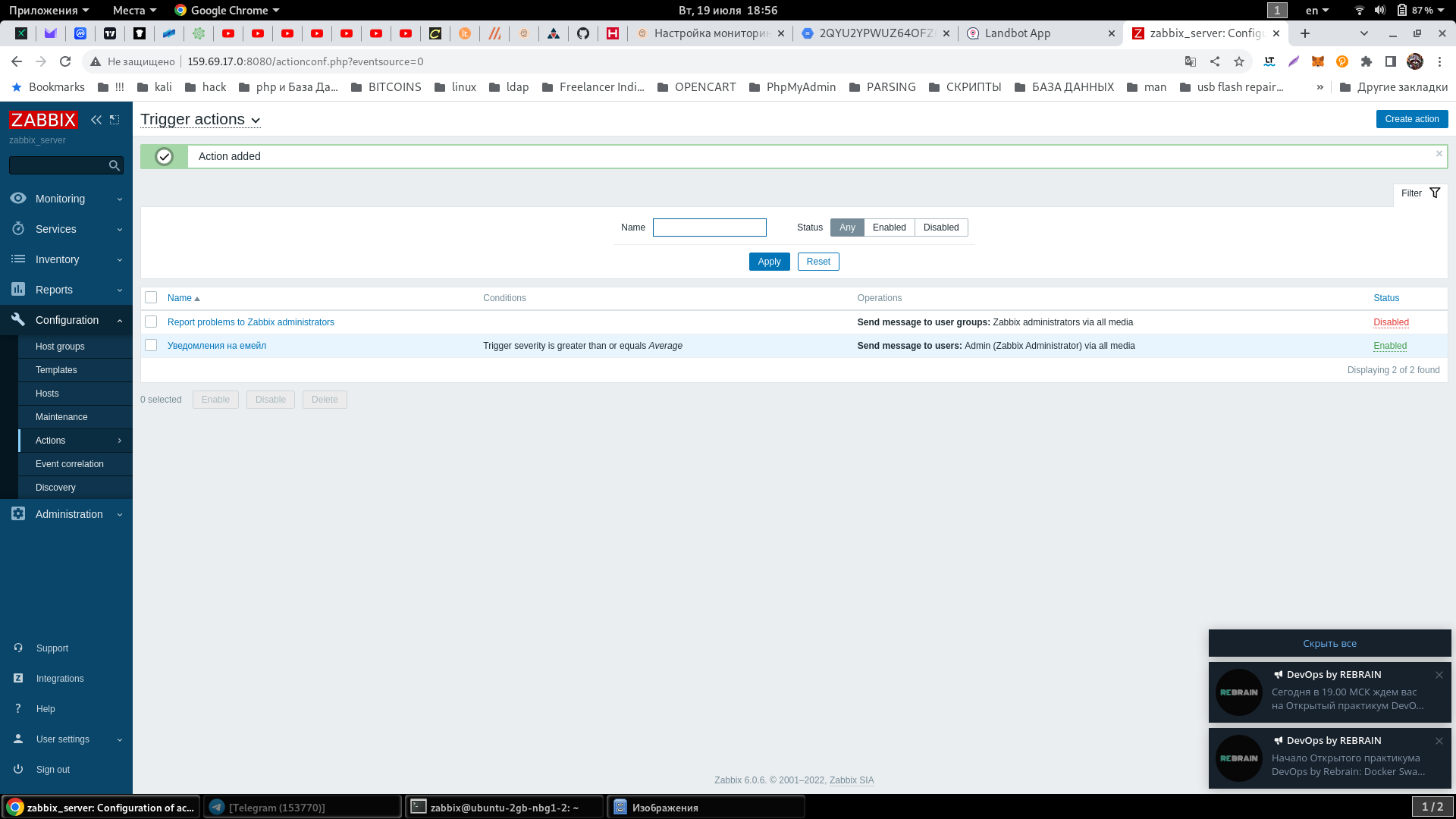
После этого нажать **Add** внутри блока **Operations.**

И только когда мы увидим одну запись в блоке Operations то можно жать основную кнопку **Add.**





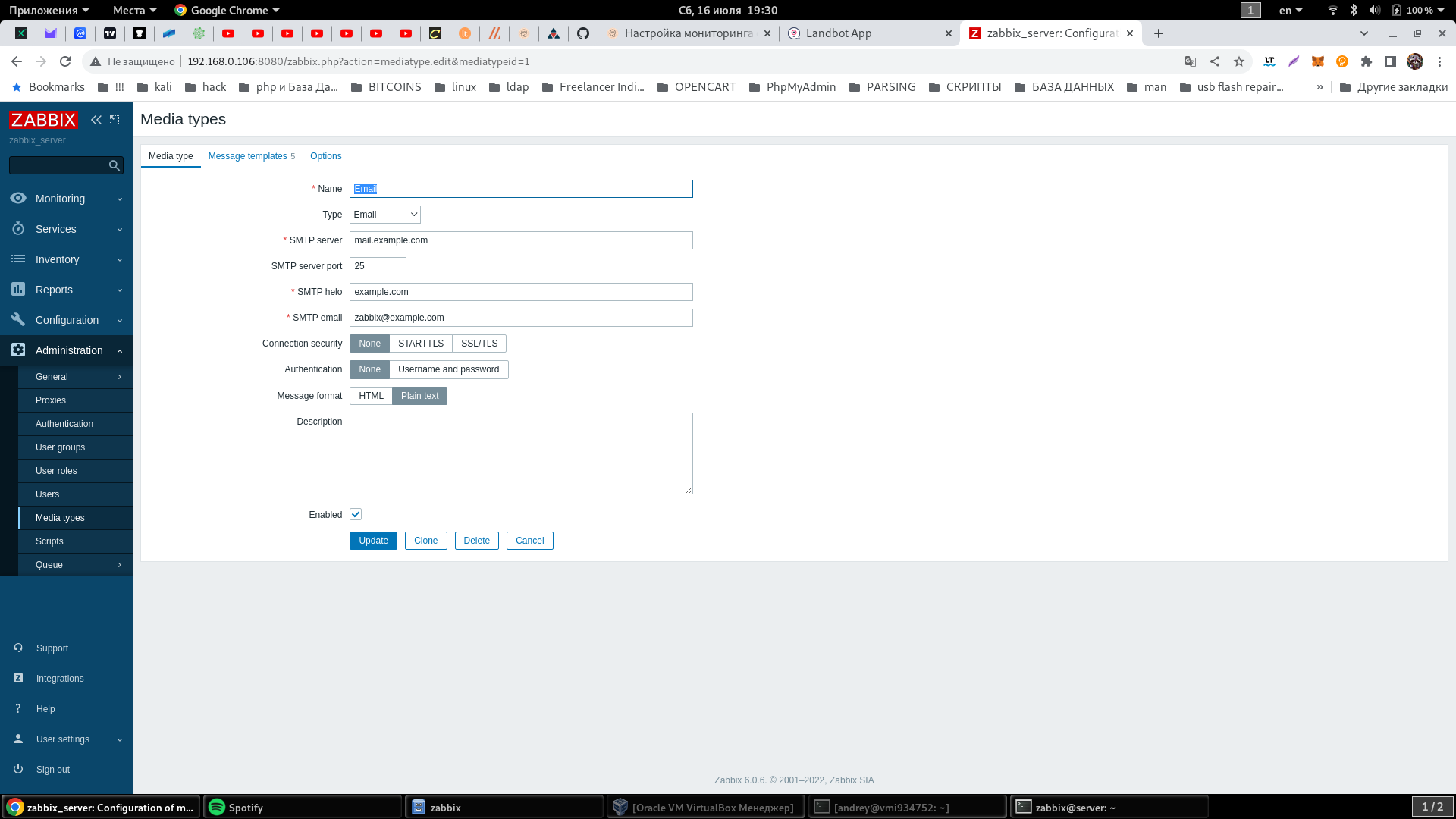




Теперь идем в **Administration - Media Types**

В этом разделе задаются настройки каналов уведомлений. Zabbix умеет слать уведомления через **Email,** либо через **скрипт**. Скрипт можно написать под любой мессенджер =). Мы этого делать пока не будем, но теоретически, если захочешь, ты можешь попытаться наладить отправку уведомлений в телеграм.

Мы же выбираем уже имеющийся тип уведомлений **Email** и видим много полей для заполнения.



В эти поля нужно вписать реальный сервер и учетные данные. Разумнее всего будет завести отдельную учетную запись под сервер мониторинга.

Если нет своего почтового сервера, то можно завести почтовый ящик на площадке гугла или яндекса.

Чтобы проверить, что мы все настроили правильно существует кнопочка test

