

OpenQA: Автоматизированное тестирование графических приложений и сценариев установки

ИНФЕРИТ
МСВСфера
ОПЕРАЦИОННАЯ СИСТЕМА

Евгений Замрий
Главный архитектор
Evgeniy.Zamriy@inferit.com
OS Day 2023

О докладчике

- Занимаюсь разработкой дистрибутивов GNU/Linux и автоматизацией, связанных с этим задач, с 2012 года
- Участвовал в разработке дистрибутивов AlmaLinux (сооснователь) и CloudLinux OS (Director of Release Engineering)
- GNU/Linux в качестве основной ОС более 25 лет
- **Текущий проект:**
 - Главный архитектор **Инферит ОС МСВСфера**
 - **Инферит** - отечественный производитель компьютерной техники, серверного оборудования, инфраструктурного обеспечения, облачных решений и сервисов
 - **МСВСфера** - операционная система общего назначения (Сервер и Рабочая станция) с 10-летним циклом поддержки

Проблемы тестирования ОС

- Дистрибутив постоянно обновляется
- Количество пакетов увеличивается
- Различные редакции и варианты установки ОС
- Множество архитектур: x86_64, aarch64, e2k, ppc64le, s390x, ...
- Ручное тестирование - дорого, долго и неэффективно
- Автоматическое тестирование:
 - Сложность в реализации
 - Требуются специалисты: тестировщики, разработчики, devops
 - Часто осуществляется с позиции разработки - скрипты, API, наборы данных, системы управления конфигурацией и т.п.
 - **Но пользователь взаимодействует с интерфейсом... Графическим!**

Инструменты тестирования ОС

- Оркестрация/gating: Jenkins, GitLab CI, Zuul, ...
- Консольные приложения: Bats, Expect, Bash, ...
- Web-приложения: Selenium, Puppeteer, Playwright, ...
- Серверные приложения/инфраструктура:
 - Ansible, Salt, Puppet, Chef
 - TestInfra, ServerSpec
 - Terraform, VMs, containers, clouds
- Установка: Kickstart (Red Hat) / Preseed (Debian), Packer + см. выше
- Графические приложения: **?!**

Решение: openQA

- Платформа для тестирования дистрибутивов
- Разрабатывается проектом openSUSE с 2009 года
- Технологии: Perl, QEMU, PostgreSQL, OpenCV, Tesseract
- Свободное ПО: GPLv2
- Ресурсы:
 - <https://open.qa/>
 - <https://github.com/os-autoinst/openQA/>



OpenQA: ключевые особенности

- Тестирование со стороны пользователя:
 - Анализ содержимого экрана с помощью OpenCV и Tesseract OCR
 - Видеозапись изображения на экране
 - Эмуляция действий пользователя с клавиатурой и мышью
 - Запись аудио и сопоставление с образцом
 - Выполнение консольных и графических приложений
 - Работа с виртуальными терминалами и управление питанием
- Возможность тестирования в окружении из нескольких систем
- Отладка: дампы памяти, снимки дисков, последовательная консоль, BIOS и UEFI режимы
- Передача файлов, работа с отчётами в форматах JUnit, XUnit, LTP

openQA: сообщество

openSUSE

- <https://openqa.opensuse.org/>
- <https://github.com/os-autoinst/os-autoinst-distri-opensuse/>

Fedora

- <https://openqa.fedoraproject.org/>
- <https://pagure.io/fedora-qa/os-autoinst-distri-fedora/>

Debian

- <https://openqa.debian.net/>

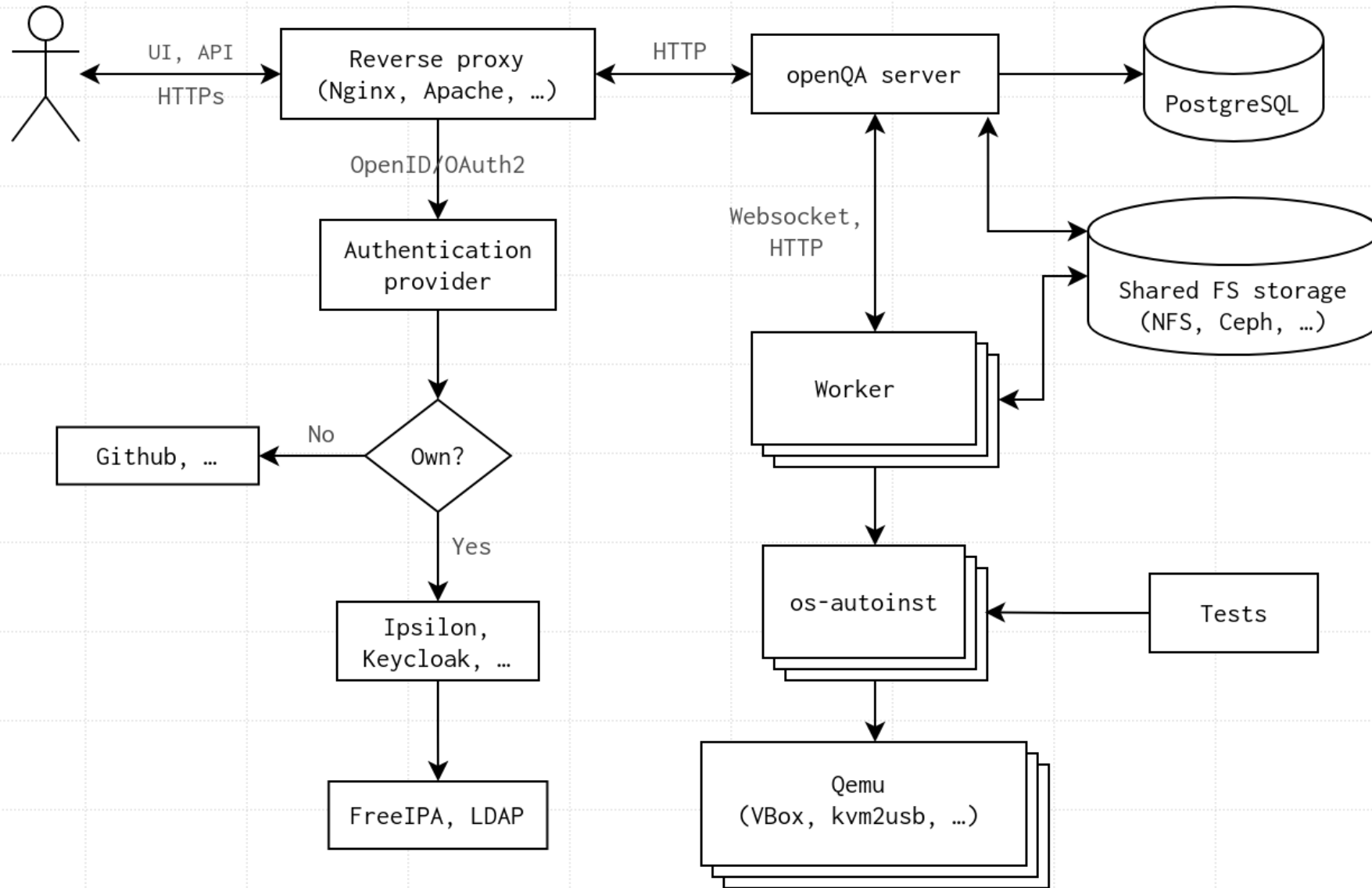
AlmaLinux

- <https://openqa.almalinux.org/>
- <https://github.com/AlmaLinux/os-autoinst-distri-almalinux/>

Другие ОС

- MCBCФера, EuroLinux, Rocky Linux, Qubes OS

openQA: архитектура



openQA: пример проекта

```
lib/
  distribution.pm
  installedtest.pm
needles/
  bootloader_bios.json
  bootloader_bios.png
tests/
  bootloader.pm
  anaconda.pm
main.pm
msvsphere-templates.json
# библиотека общих функций
# класс поддержки дистрибутива
# базовый класс для тестов
# библиотека эталонных изображений
# метаданные эталонного снимка
# эталонный снимок экрана
# каталог с тестовыми сценариями
# сценарий тестирования загрузчика
# сценарий тестирования anaconda
# точка входа - загрузчик тестов
# конфигурация тестовой среды
```

openQA: конфигурация и запуск

```
1 {
2   {
3     "JobTemplates": [
4       {
5         "test_suite_name": "boot",
6         "prio": 50,
7         "group_name": "msvsphere",
8         "machine_name": "x86_64-bios",
9         "arch": "x86_64",
10        "flavor": "everything-dvd-iso",
11        "distri": "msvsphere",
12        "version": "*"
13      }
14    ],
15    "Products": [
16      {
17        "name": "msvsphere-everything-dvd-iso-x86_64-*",
18        "flavor": "everything-dvd-iso",
19        "distri": "msvsphere",
20        "arch": "x86_64",
21        "version": "*",
22        "settings": [{"key": "TEST_TARGET", "value": "ISO"}]
23      }
24    ],
25    "TestSuites": [
26      {
27        "name": "install_gui_minimal",
28        "settings": [
29          {"key": "INSTALL", "value": "1"},
30          {"key": "PACKAGE_SET", "value": "minimal"}
31        ]
32      }
33    ]
34  }
35 }
```



```
29 "Machines": [
30   {
31     "name": "x86_64-bios",
32     "backend": "qemu",
33     "settings": [
34       {
35         "key": "ARCH_BASE_MACHINE",
36         "value": "64bit"
37       },
38       {
39         "key": "PART_TABLE_TYPE",
40         "value": "mbr"
41       },
42       {
43         "key": "QEMUCPU",
44         "value": "Nehalem"
45       },
46       {
47         "key": "QEMUCPUS",
48         "value": "2"
49       },
50       {
51         "key": "QEMURAM",
52         "value": "3072"
53       },
54       {
55         "key": "QEMU_VIDEO_DEVICE",
56         "value": "virtio-vga"
57       }
58     ]
59   }
60 ]
```

```
openqa-cli api -X POST isos ISO=MSVSpHERE-9.1-alpha-x86_64-dvd.iso DISTRI=msvsphere VERSION=9.1
FLAVOR=everything-dvd-iso ARCH=x86_64 BUILD=msvsphere-9.1-$(date +%Y%m%d%H%M)
```

openQA: main.pm

```
1 use strict; use testapi; use autotest;
2 my $distro = testapi::get_var('CASEDIR') . '/lib/MSVSphere/distribution.pm';
3 require $distro;
4 testapi::set_distribution(MSVSphere::distribution->new());
5
6 if (get_var('ISO')) {                                # набор тестов для ISO
7     autotest::loadtest('tests/bootloader.pm');
8     autotest::loadtest('tests/anaconda_install_gui.pm');
9     autotest::loadtest('tests/first_boot.pm');
10    autotest::loadtest('tests/collect_data.pm');
11    autotest::loadtest('tests/release_info.pm');
12 }
13 elsif (get_var('NETWORK')) { ... }                 # набор тестов для сетевой установки
14 else { ... }                                       # набор тестов установленной системы
```

Загрузчик
тестов

openQA: тест в терминале

```
1 use base 'MSVSphere::installedtest'; use strict; use testapi; use File::Basename;
2 # tests/collect_data.pm: собираем информацию с тестовой машины (пример работы с терминалом)
3 sub run {
4     my $self = shift;
5     $self->open_root_console(); # переключиться на root консоль
6     my $installed_path = '/var/tmp/installed_rpms.log';
7     assert_script_run("rpm -qa | sort > ${installed_path}"); # запуск скрипта на VM
8     my @to_upload = ('/etc/sphere-release', '/etc/os-release', '/etc/fstab', $installed_path);
9     foreach my $file_path (@to_upload) { # загрузить файлы в хранилище
10         upload_logs($file_path, log_name => basename($file_path));
11     }
12     send_key('ctrl-d'); # выйти из root консоли
13 }
```

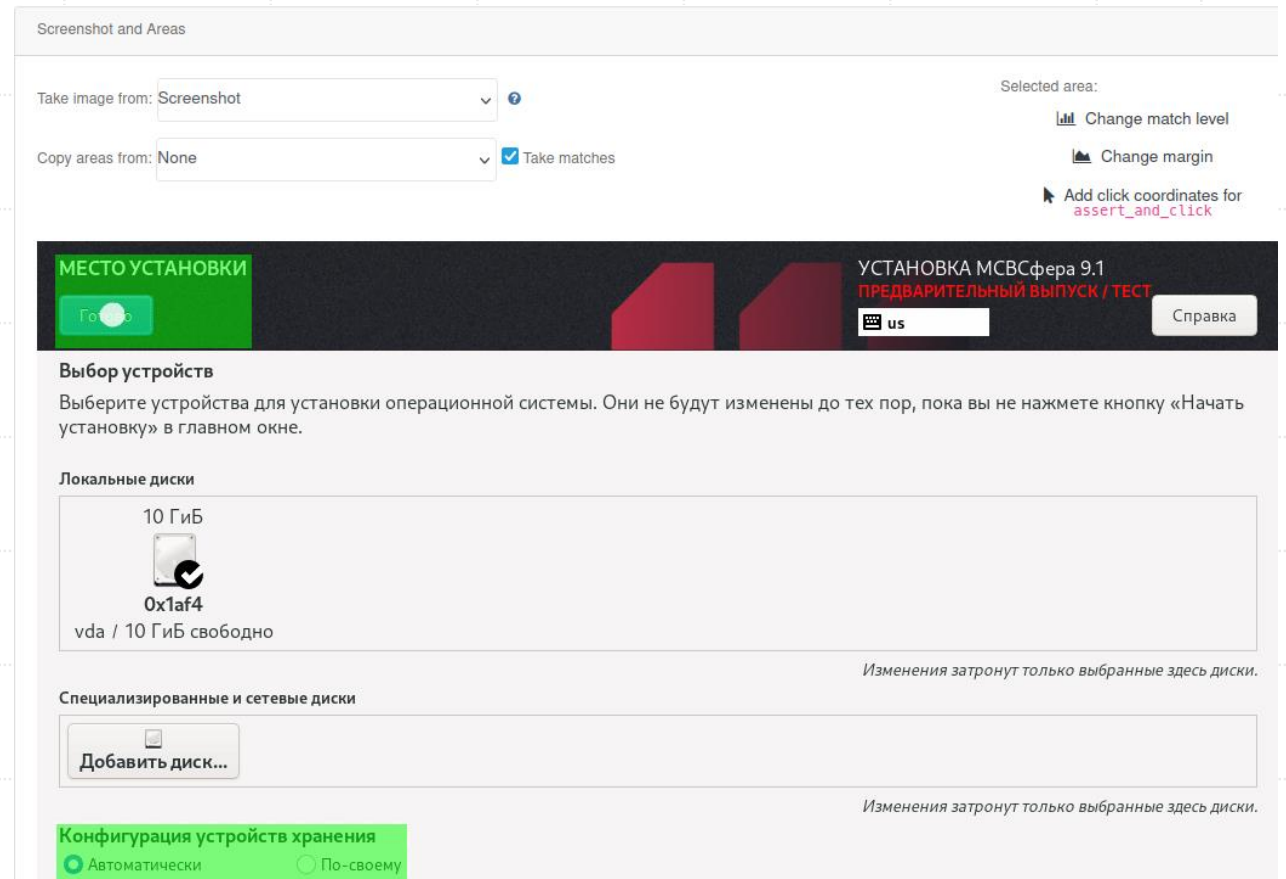
openQA: тест GUI

```
1 use base 'basetest'; use strict; use testapi;
2 # tests/anaconda_install_gui.pm: графическая установка системы
3 sub run {
4     # ожидаем совпадения с соответствующим needle и выполняем левый щелчок в указанной области
5     assert_and_click('anaconda_welcome_gui', timeout => 120);
6     if (check_var('ISO_PREVIEW_VERSION', '1')) { assert_and_click('anaconda_preview'); }
7     # устанавливаем пароль root
8     assert_and_click('anaconda_main_root_password');
9     my $root_passwd = get_var('ROOT_PASSWORD');
10    type_string($root_passwd);           # работа с клавиатурой
11    send_key('tab');
12    type_string($root_passwd);
13    assert_and_click('anaconda_root_password_confirm');
14    ...
15    assert_screen_change(sub { }, 10);   # ожидаем изменения содержимого экрана
16 }
17 # другие методы: save_screenshot, check_screen, match_has_tag, click_lastmatch, assert_and_dclick, wait_screen_change и т.д.
```

openQA: needles

Needle - эталонный снимок экрана в формате PNG и JSON метаданные к нему:

- *Описание зон:*
 - *Тип (сопоставление, OCR, игнорирование)*
 - *Координаты и границы*
 - *% совпадения*
- *Координаты нажатия*
- *Список тегов*



Screenshot and Areas

Take image from: Screenshot

Copy areas from: None

Selected area:

- Change match level
- Change margin
- Add click coordinates for `assert_and_click`

МЕСТО УСТАНОВКИ

Готово

УСТАНОВКА MSVCфера 9.1
ПРЕДВАРИТЕЛЬНЫЙ ВЫПУСК / ТЕСТ

us Справка

Выбор устройств

Выберите устройства для установки операционной системы. Они не будут изменены до тех пор, пока вы не нажмете кнопку «Начать установку» в главном окне.

Локальные диски

10 GiB

0x1af4

vda / 10 GiB свободно

Изменения затронут только выбранные здесь диски.

Специализированные и сетевые диски

Добавить диск...

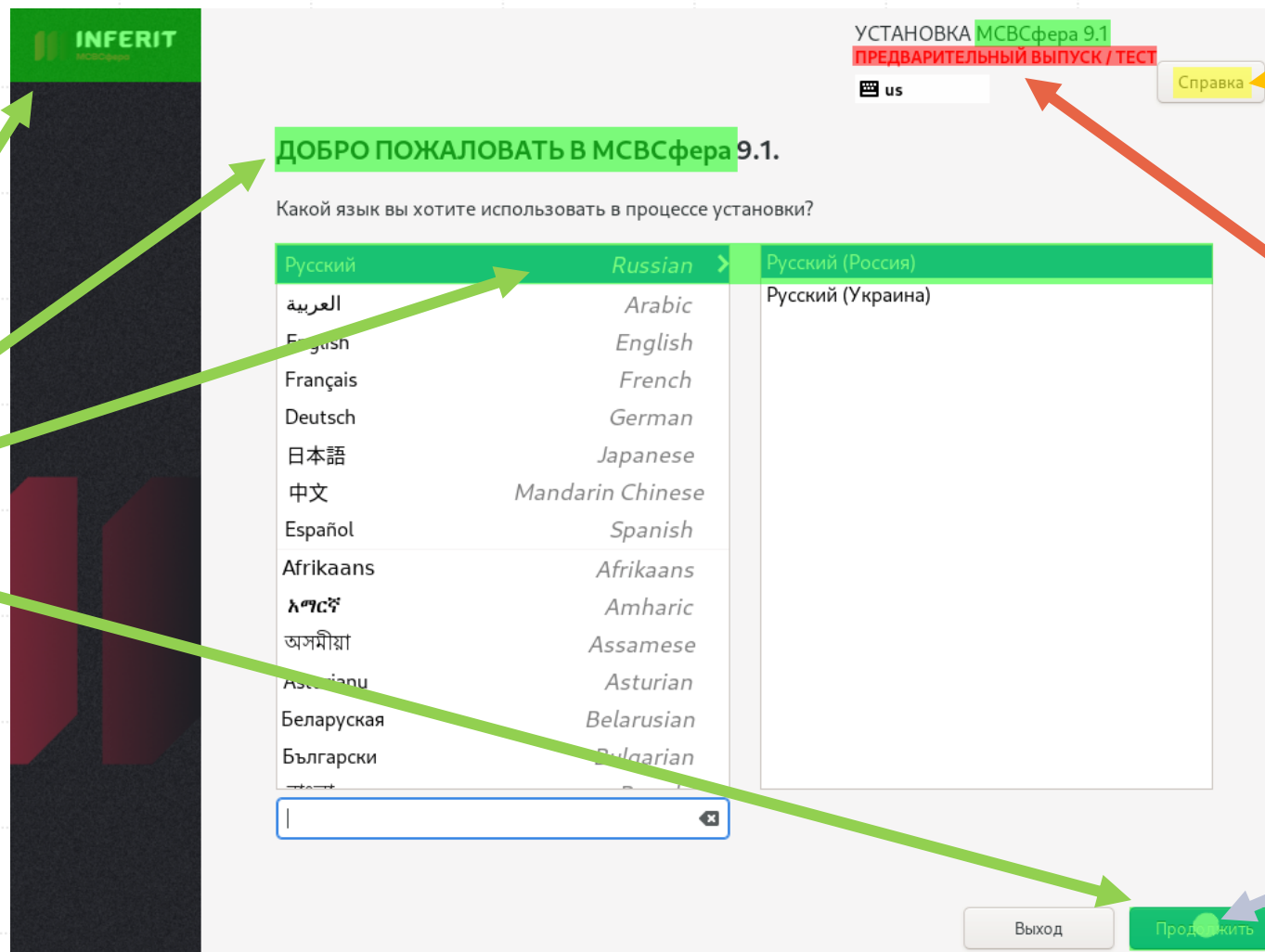
Изменения затронут только выбранные здесь диски.

Конфигурация устройств хранения

Автоматически По-своему

Встроенный редактор needle

openQA: needles, зоны



Зоны сопоставления



Зона OCR сравнения

Зона игнорирования








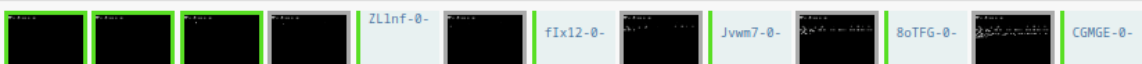

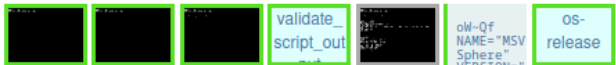
Область нажатия

openQA: тестовая задача

Results for msvsphere-9.1-everything-dvd-iso-x86_64-Buildmsvsphere-9.1-202306172333-install_minimal@x86_64-bios

Result: **passed**, finished 4 days ago (04:22 minutes)  
Scheduled product: msvsphere-9.1-everything-dvd-iso-x86_64-msvsphere-9.1-202306172333
Assigned worker: bugmob.zamry.info:1
Minimal installation using anaconda GUI

[Details](#) [Logs & Assets](#) [Settings](#) [Comments \(0\)](#) [Next & previous results](#)

Test	Result	References
tests		
bootloader 1s 	passed	
anaconda_install_gui 2m 54s 	passed	
first_boot 15s 	passed	
collect_data 45s 	passed	
release_info 22s 	passed	

```
/etc/os-release: NAME = MSVSphere
/etc/os-release: ID = msvsphere
/etc/os-release: VERSION_ID = 9.1
/etc/os-release: VERSION = 9.1 (Inferit)
/etc/os-release: ID_LIKE = rhel centos fedora
/etc/os-release: PRETTY_NAME = MSVSphere 9.1 (Inferit)
/etc/os-release: PLATFORM_ID = platform:el9
/etc/os-release: HOME_URL = https://msvsphere.ru/
/etc/os-release: BUG_REPORT_URL = https://bugs.msvsphere.ru/
/etc/os-release: CPE_NAME = cpe:/o:ncsd:msvsphere:9::baseos
/etc/os-release: REDHAT_SUPPORT_PRODUCT = MSVSphere
/etc/os-release: REDHAT_SUPPORT_PRODUCT_VERSION = 9.1
```


openQA: выводы

Достоинства

- Широкие возможности по применению
- Документация и примеры, зрелый проект
- СПО, используется лидерами индустрии
- Большое сообщество, множество открытых наработок
- Значительное ускорение выпуска релизов

Недостатки

- Высокий порог вхождения
- Проекты сообщества имеют сложную структуру
- Сложности с поиском Perl разработчиков
- Декларируется поддержка тестов на Python, но реализована через Inline::Python - возникают трудности



Спасибо за внимание!

Вопросы?

ИНФЕРИТ
МСВСфера
ОПЕРАЦИОННАЯ СИСТЕМА

Евгений Замрий
Главный архитектор
Evgeniy.Zamriy@inferit.com
OS Day 2023