

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ «КАУЧ»

BENCH ML
РУКОВОДСТВО ПО УСТАНОВКЕ И
ЭКСПЛУАТАЦИИ

для версии 2.1

ОГЛАВЛЕНИЕ

1	ИНФОРМАЦИЯ О СИСТЕМЕ	3
1.1	Системные требования.....	3
1.2	Архитектура и принципы функционирования.....	4
1.3	Лицензирование и поставка.....	5
2	УСТАНОВКА	6
3	РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ	8
3.1	Объекты	8
3.1.1	<i>Создание, редактирование и удаление объектов</i>	8
3.1.2	<i>Запуск проверок, информация о состоянии объекта</i>	14
3.2	Проверки	16
3.3	Профили.....	19

1 ИНФОРМАЦИЯ О СИСТЕМЕ

Bench ML – программное обеспечение, автоматизирующее функции мониторинга соответствия параметров конфигураций сетевого и серверного оборудования требуемым значениям.

1.1 Системные требования

Требования к программной среде представлены в Таблице 1:

Таблица 1 - Требования к программной среде

№	Характеристика	Требование
1	Операционная система	ALT Linux 10, Astra Linux 1.7/2.12, RED OS 7.3, Ubuntu 20.04 LTS или более поздняя
2	Языки программирования	Python 3.7 или более поздний
3	Средства управления базами данных	Postgres Pro 10, PostgreSQL 10 или более поздние
4	Наличие на клиентской машине, используемых в процессе функционирования	Последние версии браузеров Яндекс Браузер, Chrome или Mozilla Firefox

Требования к аппаратным характеристикам представлены в Таблице 2.

Таблица 2 - Требования к серверному оборудованию

№	Характеристика	Требование
1	Сервер	Процессор: не менее 2 ядер x 2 GHz
		Оперативная память: не менее 2 GB
		Жесткий диск: не менее 50 GB

Для бесперебойного функционирования необходимо обеспечить канал связи между сервером и клиентской рабочей станцией, а также между сервером

приложения и контролируемые серверами пропускной способностью не менее 1 Мбит/сек.

1.2 Архитектура и принципы функционирования

Системы Bench ML состоит из монолитного приложения, реализующего взаимодействие с пользователем и выполнение всех функций, и традиционной реляционной СУБД. Приложение и СУБД могут размещаться как на одном физическом или виртуальном сервере, так и на различных. Резервирование, при необходимости, осуществляется средствами СУБД, а также сторонними средствами в случае резервирования приложения.

Интерфейс для работы пользователей с системой – веб, может использоваться любой современный компьютер с последними версиями популярных браузеров.

Сервер приложений системы необходимо разместить в сегменте сети, в который имеется или организуется сетевой доступ ко всем устройствам и/или их компонентам, которые планируется контролировать. В ходе работы, система инициирует подключения со своего сервера к контролируемым ресурсам по протоколам SSH (для сетевого оборудования и серверов под управлением ОС семейств Unix и Linux) и SMB (для серверов под управлением ОС Windows) с заранее сохраненными учетными данными и выполняет проверки по заранее подготовленным командам.

По результатам проверок в интерфейсе системы отображаются детальные результаты каждой выполненной проверки и аналитическая информация, включая динамику изменений состояния контролируемых ресурсов.

1.3 Лицензирование и поставка

Системы Bench ML лицензируются по количеству серверных и сетевых единиц, которые могут контролироваться системой. Обновления функционала приобретенной системы осуществляется в рамках технической поддержки.

Поставка дистрибутива Bench ML осуществляется в электронном виде (скачивание с сайта производителя по предоставленной ссылке и с использованием предоставленных реквизитов) или на материальном носителе. Установка и настройка могут быть осуществлены производителем или его представителями, если это предусматривается отдельно согласуемыми условиями поставки.

При возникновении ошибок необходимо обратиться к производителю по предоставленным в ходе поставки контактам (при утере или недоступности можно направить соответствующий запрос по адресу info@couch.ru).

2 УСТАНОВКА

Для установки требуется сервер с развернутым дистрибутивом Ubuntu Linux.

Для запуска приложения потребуются Python 3 и Postgres Pro/PostgreSQL. Чтобы их установить, при наличии соединения с интернетом и доступности репозитория. Например, для ОС Astra Linux и Ubuntu введите следующие команды (и пароль пользователя системы, когда sudo его запросит – для ОС):

- *sudo apt update*
- *sudo apt install postgresql python python3*

Далее потребуется создать системного пользователя приложения и базу данных в СУБД Postgres, например:

- *sudo -u postgres psql*
 - *CREATE USER benchuser WITH PASSWORD 'benchpass';*
 - *CREATE DATABASE benchdb WITH OWNER benchuser;*
 - *\q*

Для начала установки, загрузите архив с приложением и распакуйте его. Для распаковки воспользуйтесь командой tar, например:

- *cd путь_до_папки_с_архивом*
- *tar xfa bench.tar.gz*

Перейдите в распакованную папку и запустите install.sh от суперпользователя:

- *cd bench*
- *sudo ./install.sh*

Далее скрипт будет запрашивать данные, которые нужно будет ввести с клавиатуры.

Таблица 3 – Установка приложения

Текст, выведенный скриптом	Пример данных, которые требуется ввести
Please, enter PostgreSQL hostname:	Введите <i>localhost</i>
Please, enter PostgreSQL database:	Введите <i>benchdb</i>
Please, enter PostgreSQL username:	Введите <i>benchuser</i> (ввод не будет отображаться на экране)
Please, enter PostgreSQL password:	Введите <i>benchpass</i> (ввод не будет отображаться на экране)

После того, как появилось сообщение «Thank you for installing Bench.», можно запустить Bench командой:

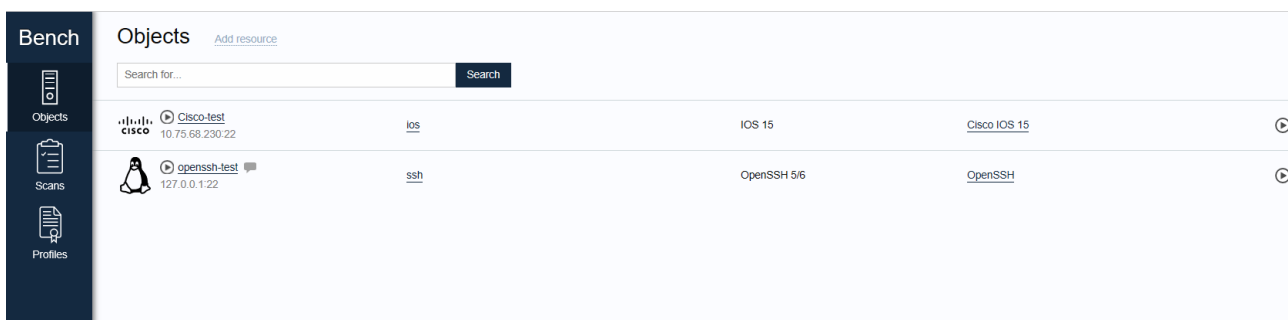
- *./bench*

Доступ к приложению можно получить в браузере на странице <http://localhost:8080>

3 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Данный раздел содержит информацию об использовании ПО Bench.

Основная страница приложения, на которой представлен перечень всех контролируемых системой объектов, выглядит следующим образом:



Средством навигации по разделам приложения служит панель, расположенная в левой части экрана.

Система позволяет оценивать заведенные в нее объекты на соответствие подготовленным профилям путем проведения проверок (сканирований). В качестве объектов системы выступают программные компоненты серверов и сетевого оборудования (которые в терминологии системы называются ресурсами).

3.1 Объекты

3.1.1 Создание, редактирование и удаление объектов

Добавление нового контролируемого ресурса осуществляется по нажатию на кнопку «Add resource»:

Add resource ×

Name: *

Comment:

Resource type:

Resource class:

Address: *

Port: *

Type: *

Login: *

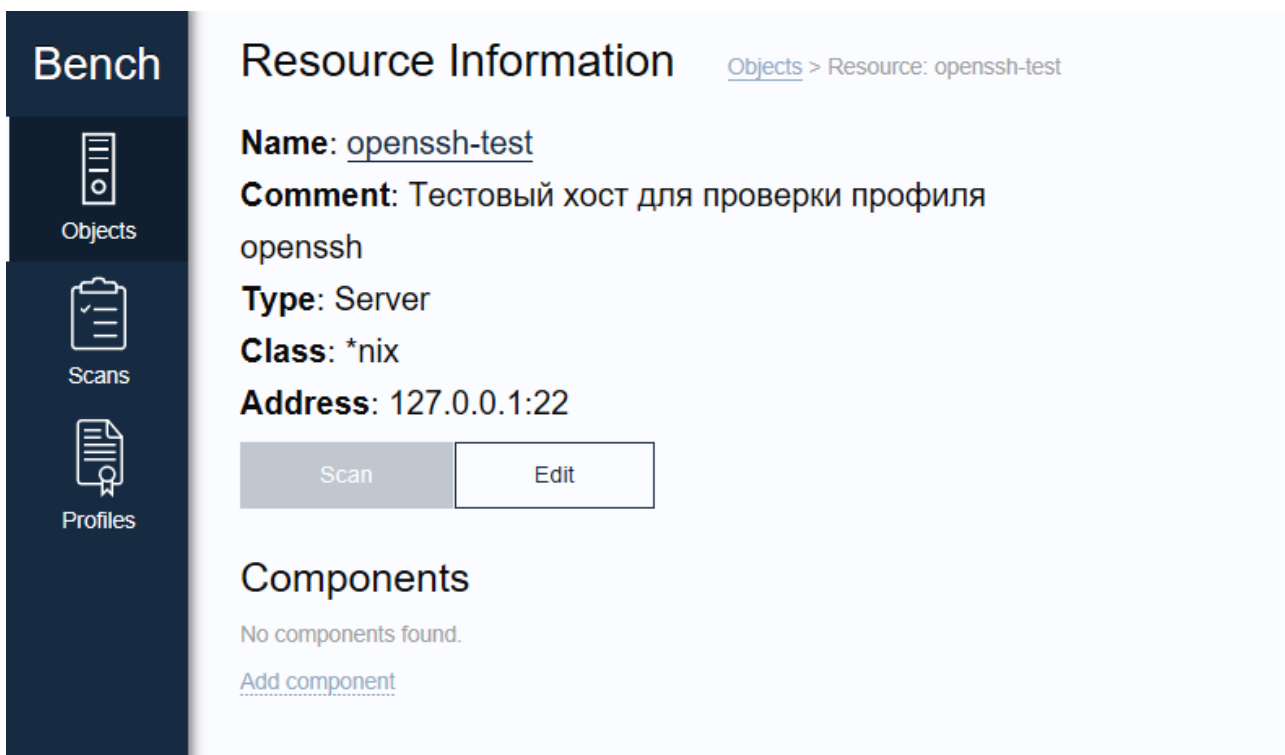
Password: *

В появившемся диалоговом окне необходимо указать:

- уникальное имя ресурса,
- комментарий (опционально),
- тип ресурса (серверное или сетевое оборудование),
- класс ресурса (семейство ОС),
- адрес ресурса,
- порт, по которому будет осуществляться доступ,

тип и соответствующие реквизиты аутентификации.

После успешного создания ресурса откроется его страница:



The screenshot displays the 'Resource Information' page in the Bench ML application. On the left is a dark sidebar with the 'Bench' logo and three menu items: 'Objects' (selected), 'Scans', and 'Profiles'. The main content area has a light blue header with the title 'Resource Information' and a breadcrumb trail 'Objects > Resource: openssh-test'. Below the header, the resource details are listed: 'Name: openssh-test', 'Comment: Тестовый хост для проверки профиля openssh', 'Type: Server', 'Class: *nix', and 'Address: 127.0.0.1:22'. There are two buttons, 'Scan' and 'Edit', below the address. At the bottom, the 'Components' section shows 'No components found.' and a link to 'Add component'.

Данная страница постоянно доступна по нажатию на имя ресурса с главной страницы (страница объектов). На странице доступна общая информация о ресурсе, указанная при его создании или измененная в процессе эксплуатации.

Далее необходимо добавить компонент, к которому можно будет привязать один из доступных профилей сканирования. Один ресурс может иметь несколько компонентов, например, соответствующих программному наполнению аппаратной единицы (для сервера – компоненты ОС, СУБД, прикладного ПО (OpenSSH, Apache, MS IIS и мн. др.)). Для добавления компонента необходимо нажать кнопку «Add component»:

×

Resource:

Name:

Comment:

Component type:

Profile:

В появившемся диалоговом окне необходимо указать:

- уникальное имя компонента,
- комментарий (опционально),
- тип компонента (конкретные ОС, СУБД, экземпляры прикладного ПО),
- соответствующий профиль из числа доступных.

После создания компонента откроется его страница, которая постоянно доступна по нажатию на имя ресурса с главной страницы (страница объектов):

Component Information [Objects](#) > [Resource: Cisco-test](#) > [Component: ios](#)

Name: [ios](#)
Comment: –
Order: Net
Family: IOS
Default profile: [Cisco IOS 15](#)

Scans

Time	Task	Profile	Status
No scans found			

На странице доступна общая информация о компоненте, указанная при его создании или измененная в процессе эксплуатации.

Отредактировать общую информацию ресурсов и компонентов можно по нажатию на кнопку «Edit», располагающуюся на страницах соответствующих объектов:

Edit resource ✕

Parent:
(Root) ▾

Name: *
openssh-test

Comment:
Тестовый хост для проверки профиля openssh

Address: *
127.0.0.1

Port: *
22

Edit credentials

Type: *
Login, password ▾

Login: *
[input field]

Password: *
[input field]

[Delete](#) Save

Edit component ✕

Resource:
openssh-test ▾

Name: *
ssh

Comment:
[input field]

Profile:
OpenSSH ▾

[Delete](#) Save

Удалить любой объект и всю связанную с ним информацию, включая результаты мониторинга, можно по нажатию на кнопку «Delete» в форме редактирования свойств. Перед удалением система запросит подтверждение:


Delete object ✕


Do you really want to delete **openssh-test**? All child components will be deleted too.

Delete

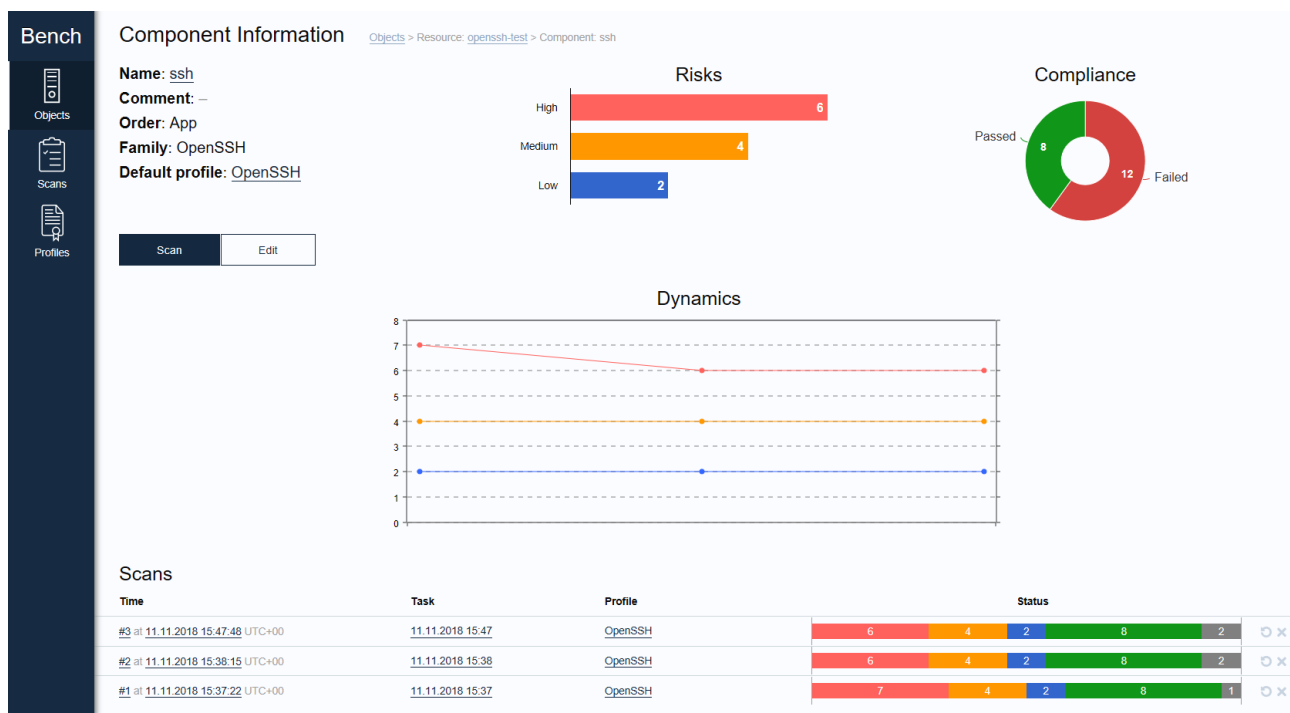
3.1.2 Запуск проверок, информация о состоянии объекта

Проверка соответствия объектов предъявляемым к ним требованиям может быть выполнена как для отдельного компонента, так и для ресурса, содержащего один или несколько компонентов.

Запустить проверку отдельного компонента можно с его страницы по нажатию на кнопку «Scan», а также в экспресс-режиме с главной страницы (страницы объектов) по нажатию на кнопку  в строке компонента.

Запустить проверку всех компонентов ресурса можно с его страницы по нажатию на кнопку «Scan», а также в экспресс-режиме с главной страницы (страницы объектов) по нажатию на кнопку  возле имени ресурса.

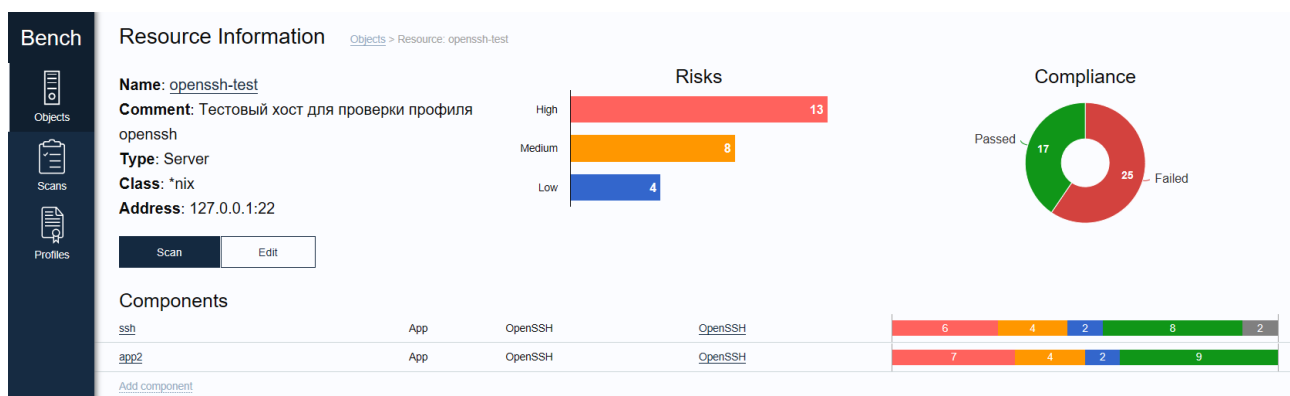
Страница компонента с произведенными сканированиями имеют следующий вид:



На странице присутствует следующая информация:

- Результаты последней проверки, отражающие текущий статус:
 - столбчатая диаграмма «Risks», на которой отмечено в разрезе уровней рисков количество требований и параметров, которые не соответствуют эталонным;
 - кольцевая диаграмма «Compliance», отражающая общее количество соответствующих и не соответствующих эталонным значениям требований и параметров;
- Диаграмму, отражающую динамику изменения результатов проверок (доступна для двух и более успешных проверок);
- Историю проверок (раздел «Scans») с краткими результатами.

Страница ресурса с произведенными сканированиями компонентов имеет следующий вид:



На странице присутствует следующая информация:

- Кумулятивные результаты последних проверок компонентов, отражающие текущий статус:
 - столбчатая диаграмма «Risks», на которой отмечено в разрезе уровней рисков количество требований и параметров, которые не соответствуют эталонным;

- кольцевая диаграмма «Compliance», отражающая общее количество соответствующих и не соответствующих эталонным значениям требований и параметров;
- Список компонентов (раздел «Components») с текущим статусом.

3.2 Проверки

Общий перечень всех выполненных проверок доступен в разделе «Scans»:

Task	Scan	Resource / Component	Status
11.11.2018 20:07	#4 at 11.11.2018 20:07:11 UTC+00	openssh-test / app2	7 (red), 4 (orange), 2 (blue), 9 (green)
11.11.2018 15:47	#3 at 11.11.2018 15:47:48 UTC+00	openssh-test / ssh	6 (red), 4 (orange), 2 (blue), 8 (green), 2 (grey)
11.11.2018 15:38	#2 at 11.11.2018 15:38:15 UTC+00	openssh-test / ssh	6 (red), 4 (orange), 2 (blue), 8 (green), 2 (grey)
11.11.2018 15:37	#1 at 11.11.2018 15:37:22 UTC+00	openssh-test / ssh	7 (red), 4 (orange), 2 (blue), 8 (green), 1 (grey)

Сводный список проверок содержит следующую информацию:

- название задания – единицы, соответствующей одной или, в случае сканирования ресурсов с несколькими компонентами, несколькими проверкам;
- номер и дату проверки;
- имена ресурса и компонента с ссылками на их страницы;
- результаты сканирования.

Для перехода к результатам конкретной проверки необходимо нажать на ссылки в поле «Scan», либо диаграмму с результатами. Также к результатам конкретной проверки можно перейти со страницы компонента, нажав там на соответствующие поля.

Вид страницы проверки:

Bench

- Objects
- Scans
- Profiles

Scan Results

Scan: #1
Time: 11.11.2018 15:37:22 UTC+00
Task: 11.11.2018 15:37
Profile: OpenSSH

Run again

Delete

Risks

High	6
Medium	4
Low	2

Compliance

Passed	8
Failed	12

Component Information

Name: ssh
Comment: –
Order: App
Family: OpenSSH
Default profile: OpenSSH

Resource Information

Name: openssh-test
Comment: Testовый хост для проверки профиля openssh
Type: Server
Class: *nix
Address: 127.0.0.1:22

# Title	Description	Remediation	Risk	Score	Value
1.1. Keep the Version Updated		Upgrade the OpenSSH service to a version that has no security bugs and apply any patches recommended by the vendor.	High		<p>[Status Not Compliant was set by on 11.11.2018 15:39:05]</p> <p>Package: openssh-server Status: install ok installed Priority: optional Section: net Installed-Size: 677 Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com> Architecture: amd64 Multi-Arch: foreign Source: openssh Version: 1:7.6p1-4ubuntu0.1 Replaces: ssh, ssh-krb5 Provides: ssh-server Depends: adduser (>= 3.9), dpkg (>= 1.9.0), libpam-modules (>= 0.72-9), libpam-runtime (>= 0.76-14), lsb-</p>

Bench

- Objects
- Scans
- Profiles


1.7. Set SSH MaxAuthTries to 4 or Less	number of authentication attempts permitted per connection. When the login failure count reaches half the number, error messages will be written to the syslog file detailing the login failure.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: MaxAuthTries 4	High		Not configured
1.8. Set SSH IgnoreRhosts to Yes	The IgnoreRhosts parameter specifies that .rhosts and .shosts files will not be used in RhostsRSAAuthentication or HostbasedAuthentication.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: IgnoreRhosts yes	High		Default setting is fine
1.9. Set SSH HostbasedAuthentication to No	The HostbasedAuthentication parameter specifies if authentication is allowed through trusted hosts via the user of .rhosts, or /etc/hosts equiv, along with successful public key client host authentication. This option only applies to SSH Protocol Version 2 .	Edit the /etc/ssh/sshd_config file to set the parameter as follows: HostbasedAuthentication no	High		Default setting is fine
1.10. Disable Rhosts Use		Change the following lines in sshd_config as described. RhostsAuthentication no; RhostsRSAAuthentication no	High		_#f
1.11. Disable TCP Connection Forwarding		Change the file sshd_config to contain: AllowTcpForwarding no	Medium		Not configured
1.12. Disable SSH Root Login	The PermitRootLogin parameter specifies if the root user can log in using ssh(1). The default is no.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: PermitRootLogin no	High		Not configured
1.13. Set SSH PermitEmptyPasswords to No	The PermitEmptyPasswords parameter specifies if the server allows login to accounts with empty password strings.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: PermitEmptyPasswords no	High		Default setting is fine
1.14. Do Not Allow Users to Set Environment Options	The PermitUserEnvironment option allows users to present environment options to the ssh daemon.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: PermitUserEnvironment no	High		Default setting is fine
1.15. Inactive Sessions		Change the sshd_config file to include the option: TCPKeepAlive yes	Medium		Default setting is fine
1.16. Reduce the allowed time to enter the password.		Change the file sshd_config to contain: LoginGraceTime 45	Low		Not configured
1.17. Use Only Approved Ciphers in Counter Mode	This variable limits the types of ciphers that SSH can use during communication.	Edit the /etc/ssh/sshd_config file to set the parameter as follows: Ciphers aes128-ctr,aes192-ctr,aes256-ctr	High		


The two options ClientAliveInterval and ClientAliveCountMax control the timeout of ssh sessions. When the ClientAliveInterval variable is set

Страница проверки содержит следующую информацию:

- Общая информация:

-
- номер сканирования;
 - время сканирования;
 - задание;
 - профиль, использованный при сканировании;
 - информация о компоненте, для которого производилась проверка;
информация о ресурсе;
 - результаты проверки:
 - Title – номер и названия проверяемого требования;
 - Descriptions – описание требования и параметров;
 - Remediations – рекомендации по настройке для соответствия;
 - Risk – критичность требования;
 - Score – оценка требования, отражающая соответствие (галка на зеленом фоне) или несоответствие (крест на красном фоне) эталонным значениям (для ручных проверок с шестеренкой на сером фоне необходимое значение должно быть выставлено вручную по нажатию ссылки «Edit»);
 - Value – вывод результатов команд, выполняемых для проверки конкретного требования, на основании которых была выставлена оценка.

Произведенная проверка может быть удалена вместе с результатами. Для этого необходимо нажать кнопку «Delete» на странице проверки или кнопку  на странице сводных проверок раздела «Scans» или на странице компонента. После подтверждения намерения в диалоговом окне, проверка будет удалена, статистическая информация компонентов и ресурсов автоматически обновится.

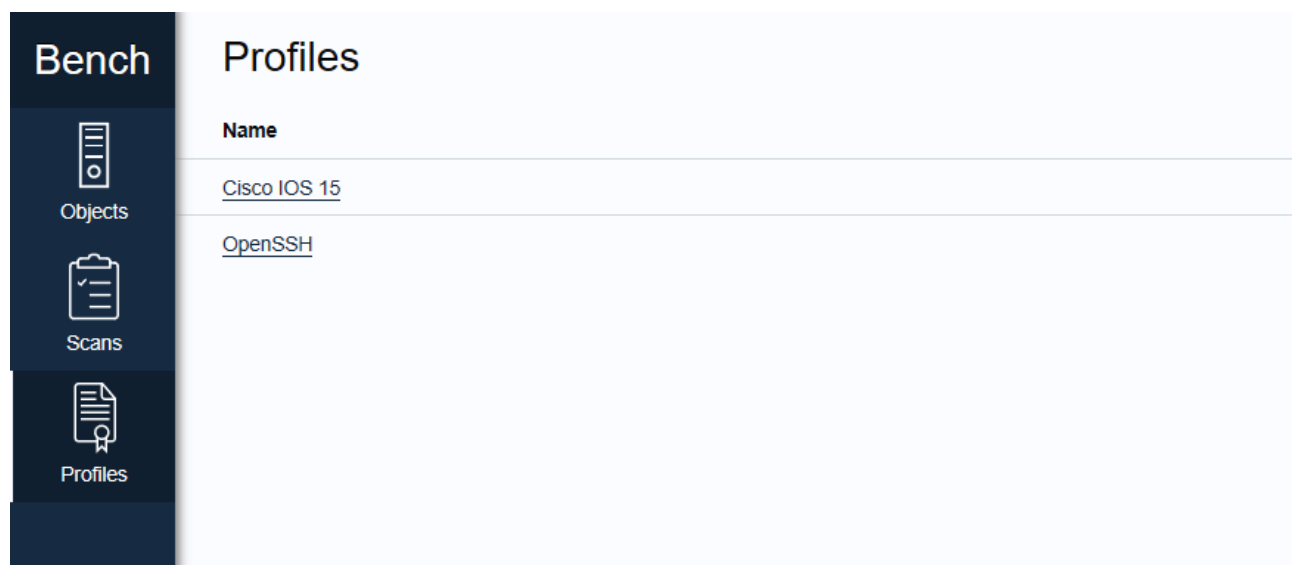
Для быстрого повторного запуска проверки можно нажать кнопку «Run again» на странице проверки или кнопку  на странице сводных проверок раздела «Scans» или на странице компонента.

3.3 Профили

Профили проверки – перечни требований и эталонных параметров, на соответствие которым проверяются конфигурации контролируемых объектов. Для каждого требования и параметра профиль содержит номер, название, описание, разъяснение, процедуры проверки и настройки, метод проверки и уровень риска.

Профили подготавливаются и поставляются в рамках поставки продукта или технической поддержки производителем. В демо-версии доступны профили проверки безопасности компонентов OpenSSH и Cisco IOS 15, создание на основе стандартов безопасного конфигурирования CIS (<https://cisecurity.org>).

Страница профилей доступна из главного меню, на ней отображаются все имеющиеся в системе профили:



Содержимое профилей доступно на странице профиля, которая открывается по нажатию на ссылку с именем профиля:

Bench		Profile Information Profiles > OpenSSH				
<ul style="list-style-type: none"> Objects Scans Profiles 		<p>Name: OpenSSH</p> <p>Comment: –</p> <p>Remarks: –</p> <p>Component Type: OpenSSH 5/6</p>				
Title	Description	Audit	Remediation	Method	Risk	
1.1. Keep the Version Updated		rpm -q openssh dpkg --status openssh-server pkg info ssh ssh -v	Upgrade the OpenSSH service to a version that has no security bugs and apply any patches recommended by the vendor.	Manual	High	
1.2. Use authentication based on public / private keys		grep ^PubkeyAuthentication /etc/ssh/sshd_config grep ^AuthorizedKeysFile /etc/ssh/sshd_config Default authorized keys location is ~/ssh/authorized_keys.	Use authentication based on public / private keys	Auto	Medium	
1.3. Set SSH Protocol to 2	SSH supports two different and incompatible protocols: SSH1 and SSH2. SSH1 was the original protocol and was subject to security issues. SSH2 is more advanced and secure.	To verify the correct SSH setting, run the following command and verify that the output is as shown: # grep "Protocol" /etc/ssh/sshd_config Protocol 2	Edit the /etc/ssh/sshd_config file to set the parameter as follows: Protocol 2	Auto	High	
1.4. Set LogLevel to INFO	The INFO parameter specifies that record login and logout activity will be logged.	To verify the correct SSH setting, run the following command and verify that the output is as shown: # grep "LogLevel" /etc/ssh/sshd_config LogLevel INFO	Edit the /etc/ssh/sshd_config file to set the parameter as follows: LogLevel INFO	Auto	Medium	
1.5. Set Permissions on /etc/ssh/sshd_config	The /etc/ssh/sshd_config file contains configuration specifications for sshd. The command below sets the owner and group of the file to root.	Run the following command to determine the user and group ownership on the /etc/ssh/sshd_config file. # /bin/ls -l /etc/ssh/sshd_config -rw-r--r-- 1 root root 762 Sep 23 00:2 /etc/ssh/sshd_config	If the user and group ownership of the /etc/ssh/sshd_config file are incorrect, run the following command to correct them: # chown root:root /etc/ssh/sshd_config If the permissions are incorrect, run the following command to correct them: # chmod 644 /etc/ssh/sshd_config	Auto	High	
1.6. Disable SSH X11 Forwarding	The X11Forwarding parameter provides the ability to tunnel X11 traffic through the connection to enable remote graphic connections.	To verify the correct SSH setting, run the following command and verify that the output is as shown: # grep "X11Forwarding" /etc/ssh/sshd_config X11Forwarding no	Edit the /etc/ssh/sshd_config file to set the parameter as follows: X11Forwarding no	Auto	Medium	

Страница профиля содержит следующую информацию:

- Общая информация профиля:
 - Name – имя профиля;
 - Comments и Remark – комментарии и особые указания для профиля;
 - Component Type – тип компонента, для которого можно использовать данный профиль;
- Требования и параметры:
 - Title – номер и названия проверяемого требования;
 - Descriptions – описание требования и параметров;
 - Audit – способ ручной проверки требования;

- Remediations – способ ручной настройки требования;
- Method – способ оценки соответствия требования:
 - Auto – требование оценивается приложением Bench автоматически,
 - Manual – требование оценивается вручную.
- Risk – критичность требования.