Обнаружение

Обнаружение

Маршрут: `/discovery`. Вкладки **Результаты** | **Правила** | **Действия**. **Новое правило** — мастер **Тип** → **Параметры** → **Действия**.

На вкладке **Действия** — каталог по типу правила (оборудование / JMX / IPMI / Server): **системное** действие (скан → Результаты) и ваши действия. Тумблеров «вкл/выкл» нет: какое действие выполняется, выбирается **на правиле**. Кнопка **Создать действие** открывает редактор; после публикации действие появляется в каталоге и на шаге **Действия** визарда правила.

На **Правила**: колонки **Имя** | **Тип** | **Область** | **Расписание** | **Статус** | **Последний запуск** | **Найдено**; фильтр типа (`network` / `jmx` / `ipmi` / `server` / `agent`); клик по строке открывает редактор. В каждой категории сети (**сеть / JMX / IPMI / Server**) одновременно может быть **только одно включённое** правило — при включении другого текущее выключается (как **одно активное** у агентов).

Обнаружение ищет то, что **сервер wiSLA может опросить сам** (ICMP, SNMP, JMX, IPMI, Server), плюс правила **Агентов** при регистрации. Уже мониторимые хосты в Результатах **показываются** со статусом «На мониторинге».

Сценарий

1. Создать **правило**: **Новое правило** → тип → визард **Параметры** → **Действия** → **Сохранить**.
   - **Область**: CIDR / диапазон / несколько через запятую (≤1024 адресов); под полем — популярные подсети и счётчик адресов.
   - **Расписание**: интервал ≥1h; пресеты `1h`…`1w`.
   - **SNMP (оборудование)** / **JMX** / **IPMI** / **Server**: на шаге **Действия** выберите **одно** действие для этого правила (системное или опубликованное своё). **Опытный режим** сохраняет черновик правила и возвращает на этот шаг после создания действия. На категорию — **одно включённое** правило.
   - **Агенты мониторинга**: без подсети; **одно активное** правило; триггер **при регистрации**; опционально **периодическая сверка** 12ч/1д/7д; два стандартных действия — **ОС после регистрации** или **ОС + поиск приложений** (READY сразу в мониторинг, CREDS — кандидаты в **Результаты**). История онбординга — **Запуски** на правиле.
   - Ошибки полей показываются под полем после **Далее** / ухода с поля; неверный формат области/интервала — сразу.
2. Каталог действий — вкладка **Действия** (`/discovery?tab=scenarios`); **Опытный режим** в мастере правила — пустой холст с возвратом в визард.

3. Запустить **Сканировать** (или дождаться расписания) — прогресс на странице, UI не зависает.

Готовые старты в редакторе действий («С чего начать»):

Пресет	Назначение
Обнаружение сетевого оборудования	Скан SNMPv2 → **Результаты** (без автопостановки)
JMX / Java	Скан JMX → **Результаты**
IPMI / BMC	Скан IPMI → **Результаты**
Server	Скан Server + ICMP → **Результаты**

Автопостановка в мониторинг — только если на холсте явно добавить блок **В мониторинг**.

На шаге **Действия** правила отдельно живёт стандарт **скан → Результаты** (без автопостановки). Свой сценарий появляется в правиле только после **публикации**.

Реакция **Ничего не делать** (`noop`) — явный пропуск ветки на холсте (узел не создаётся). Удобно для Case «иначе».

Палитра слева — три шага: **1. Сканирование** (что проверять), **2. Условие** (ветки Да/Нет или **Case** с несколькими выходами), **3. Реакция** (теги, шаблон, в мониторинг). Несколько блоков «Сканирование» в одном сценарии **складывают проверки в один проход** — второй блок не запускает второй обход сети. На холсте над схемой — краткая подсказка по сборке.

Готовые блоки условий: **Опрос ответил** / **Открыт порт** / **Описание содержит** / **Имя содержит** / **Своё условие** / **Case**. Реакция: **Теги** / **Шаблон** / **В группу** / **В мониторинг** / **Ничего не делать** / **Своя реакция**.

Case — именованные ветки плюс **Иначе**: срабатывает первая совпавшая. Внутри ветки можно несколько признаков через **или**. Для простого «да/нет» достаточно обычного условия.

Если реакция **привязывает шаблон**, которому нужны креды (HTTP, iLO, ODBC/БД и т.п.), в инспекторе под выбором шаблона появляются поля макросов (подсказка «?» у каждого). Их значения пишутся на узел при постановке. Поля с плейсхолдером `{IP}` заблокированы: при постановке подставляется адрес найденного хоста. Секреты (токен, пароль) один раз задаёте в сценарии — на все хосты ветки. SNMP community / SNMPv3 по-прежнему в **триггере скана** и уходят на SNMP-интерфейс узла, а не в макросы.

На вкладке **Действия** в колонке **Действия** — **Редактировать** (карандаш) и **Удалить**. Привязка к правилу задаётся в визарде правила, не тумблером в таблице. В триггере для действия доступны проверки **ICMP / SNMPv2 / SNMPv3 / JMX / IPMI / Server**.

Холст совпадает с воркфлоу уведомлений: **Delete** / **Backspace**, рамка мышью, **Shift**, панорама, замок холста и блока. Новый блок из палитры появляется в видимой области. Повторная карточка пресета **добавляет** сценарий ниже, не затирая уже лежащие блоки.

Просмотреть найденные хосты во вкладке **Результаты**. Строки сгруппированы по правилу. Сеть (SNMP/JMX/IPMI/Server) — как раньше. Для агентов с действием **ОС + поиск приложений** сюда же попадают **кандидаты приложений с кредами** (группа = правило агентов); Accept открывает диалог учёток и создаёт сервис `Hostname + приложение`. Приложения без кредов (READY) в Результаты не ставятся — сервис создаётся сразу при регистрации. 4. **В мониторинг** (Accept) — создать узел в инвентаре. Если SNMP ответил — **SNMP-интерфейс** с кредами. Если JMX ответил — **JMX-интерфейс** + макросы. Если IPMI ответил — **IPMI-интерфейс** + теги/макросы кредов. Если Server — диалог: evidence fingerprint + несколько HTTP/ODBC-шаблонов и макросы с `{IP}` → узел **Server** без SNMP/JMX/IPMI. Если

кандидат приложения агента — диалог кредов шаблона. 5. **Игнор** / **Удалить** — сохраняются на сервере.

Фильтр состояния **Любой** показывает все статусы, включая **Потерян**. В колонке **IP** всегда виден адрес; открытые порты — под ним. Результаты привязаны к **правилам сети**: при удалении правила связанные найденные объекты удаляются.

Состояния результата

Статус	Значение
Новый	Хост прошёл проверки сценария, ещё не принят в инвентарь
Активен	Снова отвечает после потери (ещё не принят)
На мониторинге	Уже принят или IP совпадает с узлом в инвентаре
Потерян	Раньше проходил проверки, на повторном скане — нет
Игнор	Скрыт оператором, сканы не меняют статус

Хосты, уже принятые в мониторинг или с тем же IP в инвентаре, **остаются** в Результатах со статусом **На мониторинге** (повторный scan обновляет `last_seen`).

При Accept (или операции `enableMonitoring`) применяются совпавшие операции: группа, шаблон(ы) по имени, теги, имя узла, макросы. Если hostname нет — **видимое имя = IP** (как в wiSLA discovery). Если группа не задана — `Discovered hosts`.

После **Сканировать** на вкладке **Результаты** показывается блок **Последний скан**: сколько ответило, сколько сценарий поставил на мониторинг, сколько осталось в очереди, сколько уже было в инвентаре, сколько ошибок. Если сценарий с ICMP сразу ставит всех в мониторинг, таблица кандидатов может быть пустой — это нормально: empty-state объясняет, что узлы уже поставлены, и даёт ссылку на список узлов. Тост после скана тоже отражает постановку, а не только «найдено N».

На списке сценариев и на **Результаты** видно режим: **Сразу в мониторинг** (`enableMonitoring` в реакции) или **Только очередь**. Сводка сверху считает **активные/всего сценарии**, а не устаревшие flat-actions.

Каталог «инфраструктура»: ветка **Иначе** (только ping) кладёт узлы в группу `Discovered ping-only`. Перед публикацией, если у Proxmox/iLO пустой токен или пароль, редактор предупреждает; при Accept такой шаблон не привязывается (ICMP/SNMP остаются). При ошибке скана тост показывает реальную причину с сервера, а не только подсказку про диапазон.

Технические детали: Discovery UI, Discovery API.