INVENQOR AGENT · OFFICIAL GUIDE

사용자 가이드

Linux 자산 수집 에이전트의 안전한 설치, 최초 설정, 상태 확인과 일상 사용

대상 버전	Agent v0.2.3 · Server v0.2.5
문서 버전	1.0
기준일	2026-07-29
문서 등급	공개

이 문서의 목적

이 가이드는 Invenqor Agent를 한 대의 Linux 서버에 설치하고 정상 동작을 확인해야 하는 사용자와 현장 운영자를 위한 문서입니다. 중앙 게이트웨이 설계, 인증서 수명주기, 대량 배포와 상세 데이터 사전은 관리자 가이드를 참조하십시오.

이 문서를 마치면 다음 작업을 수행할 수 있습니다.

1. 서버 CPU에 맞는 패키지를 선택하고 체크섬을 검증합니다.
2. 에이전트를 설치하고 게이트웨이 주소와 인증을 설정합니다.
3. 서비스 상태, 로그, 수집 결과와 전송 대기열을 확인합니다.
4. 기본적인 장애를 구분하고 안전하게 제거합니다.

> Invenqor Agent v0.2.3는 Server v0.2.5와 중앙 관리 콘솔을 함께 사용합니다. 서버 설치와 수집 데이터 처리 원칙은 Server 설치 및 운영 가이드를 참조하십시오.

1. 제품 이해하기

Invenqor Agent는 Linux의 `/proc`, `/sys`, `/etc`에 있는 운영체제 정보를 읽어 자산 스냅샷을 만듭니다. 수집은 외부에서 서버로 접속하는 방식이 아니라, 에이전트가 설정된 HTTPS 게이트웨이로 결과를 보내는 **outbound-only** 방식입니다.

기본 수집 범위는 다음과 같습니다.

영역	대표 수집 내용
시스템	호스트명, 배포판, 커널, 아키텍처, 부팅 시각, 시간대
CPU·메모리	논리 CPU 수, CPU 모델, load average, `/proc/meminfo` 메모리 지표
파일시스템	장치, 마운트 지점, 파일시스템, 옵션, 용량과 inode 사용량
네트워크	인터페이스, MAC/IP, 상태, MTU, 기본 경로, DNS, 로컬 포트
프로세스	PID, 이름, 상태, 부모 PID, UID/GID, 실행 파일 경로
소프트웨어	dpkg, apk 또는 rpm으로 확인한 설치 패키지
서비스	systemd, OpenRC 또는 SysV 서비스와 활성화 관련 상태
계정	사용자명, UID/GID, 홈, 셸, 보조 그룹
컨테이너	런타임 소켓, 컨테이너 내부 실행 여부, cgroup 버전

프로세스 명령행은 비밀번호나 토큰을 포함할 수 있으므로 기본값에서는 수집하지 않습니다. 파일 내용, 사용자 비밀번호 해시, 프로세스 환경변수, 원격 명령 실행 결과도 수집하지 않습니다.

2. 설치 전 준비

2.1 지원 환경

- CPU: `x86_64` 또는 `aarch64`
- 운영체제: Linux
- 검증 기준: CentOS 7, Red Hat UBI 8/9, Ubuntu 22.04/24.04 LTS, Alpine
- 권장 기준: Kernel 3.10 이상, RHEL 계열 7 이상, Ubuntu 22.04 이상, Debian 10 이상
- 호환 목표: Alpine, Amazon Linux, SUSE
- 서비스 관리자: systemd, OpenRC 또는 SysV init
- 네트워크: Server의 단일 HTTPS TCP 7070 outbound

초기 폐쇄망 설치에서 `http://사설IP:7070`, localhost 또는 내부 DNS/Kubernetes 서비스명을 사용하면 별도 옵션 없이 URL만으로 연결할 수 있습니다. 신뢰 경계를 넘는 트래픽은 반드시 HTTPS를 사용하십시오.

오래된 커널과 배포판은 핵심 `/proc` 수집만 가능할 수 있습니다. 정적 바이너리는 외부 언어 런타임을 요구하지 않지만, 실제 사용하는 시스템 호출보다 오래된 커널의 실행을 보장하지는 않습니다.

2.2 CPU 아키텍처 확인

```
uname -m
```

출력	받을 파일
`x86_64`	`invenqor-agent-linux-x86_64.tar.gz`
`aarch64` 또는 `arm64`	`invenqor-agent-linux-aarch64.tar.gz`

표에 없는 아키텍처에서는 제공된 바이너리를 실행하지 마십시오.

2.3 준비할 정보

설치 전에 관리자에게 다음 정보를 받습니다.

- 게이트웨이 기본 URL (예: `https://inventory.example.internal`)
- 인증 방식: URL-only 자동 등록(기본), 보호망용 enrollment token 또는 예외 장비별 bearer token/mTLS PEM
- 사설 인증기관을 사용한다면 CA 인증서 파일
- 조직에서 정한 수집 주기와 프로세스 명령행 수집 정책

게이트웨이가 아직 없다면 URL 없이 설치할 수 있습니다. 수집 결과는 로컬 큐에 보존되지만 기본 100 MiB 한도에 도달하기 전에 게이트웨이를 설정하거나 운영 정책을 결정해야 합니다.

3. 패키지 받기와 검증

GitHub 릴리즈 페이지에서 아키텍처에 맞는 `.tar.gz`와 같은 이름의 `.sha256` 파일을 같은 디렉터리에 받습니다.

```
curl -LO https://github.com/hkjang/invenqor-agents/releases/download/v0.2.3/invenqor-agent-linux-x86_64.tar.gz
curl -LO https://github.com/hkjang/invenqor-agents/releases/download/v0.2.3/invenqor-agent-linux-
x86_64.tar.gz.sha256
sha256sum -c invenqor-agent-linux-x86_64.tar.gz.sha256
```

정상이면 다음과 같이 표시됩니다.

```
invenqor-agent-linux-x86_64.tar.gz: OK
```

> **검증 실패 시 중단:** `FAILED`가 표시되면 압축을 풀거나 실행하지 마십시오. 파일을 삭제하고 신뢰할 수 있는 네트워크에서 다시 받으십시오.

4. 설치

4.1 압축 해제

x86_64 예시:

```
tar -xzf invenqor-agent-linux-x86_64.tar.gz
cd invenqor-agent-linux-x86_64
```

aarch64에서는 디렉터리 이름도 `invenqor-agent-linux-aarch64`입니다.

패키지는 다음 구성으로 되어 있습니다.

```
bin/invenqor-agent              실행 파일
config/config.toml              안전한 기본 설정
scripts/install.sh              설치 스크립트
scripts/uninstall.sh            제거 스크립트
service/invenqor-agent.service  systemd 정의
service/invenqor-agent.openrc   OpenRC 정의
service/invenqor-agent.init     SysV init 정의
README.md                       제품 설명
```

4.2 설치 스크립트 실행

```
sudo ./scripts/install.sh
```

스크립트는 다음 작업을 수행합니다.

- 비로그인 시스템 계정과 그룹 `invenqor-agent` 생성
- 실행 파일을 `/opt/invenqor-agent/bin/invenqor-agent`에 설치
- 최초 설치일 때만 `/etc/invenqor-agent/config.toml` 생성
- 상태 디렉터리 `/var/lib/invenqor-agent` 생성
- 발견한 init 시스템에 서비스를 등록하고 즉시 시작

기존 설정 파일은 덮어쓰지 않습니다. 재설치 후에도 설정과 미전송 큐는 유지됩니다.

4.3 설치 결과 확인

systemd:

```
sudo systemctl status invenqor-agent --no-pager
sudo journalctl -u invenqor-agent -n 50 --no-pager
```

OpenRC:

```
sudo rc-service invenqor-agent status
```

SysV init:

```
sudo service invenqor-agent status
```

버전 확인:

```
/opt/invenqor-agent/bin/invenqor-agent --version
```

예상 출력은 `invenqor-agent 0.2.3` 입니다.

5. 최초 설정

설정 파일은 TOML 형식이며 알 수 없는 키가 있으면 실행을 거부합니다. 수정 전에 백업하고, 비밀값이 포함된 파일을 메신저나 이슈에 첨부하지 마십시오.

```
sudo cp -a /etc/invenqor-agent/config.toml \
  /etc/invenqor-agent/config.toml.before-change
sudoedit /etc/invenqor-agent/config.toml
```

5.1 자동 등록 예시

```
[server]
url = "https://inventory.example.internal"
timeout_seconds = 30
```

기본 Server에서는 URL만 설정하면 Agent가 최초 전송 전에 자동 등록하고 장비 전용 Token을 상태 디렉터리에 `0600`으로 저장합니다. 이후 설정 파일에 장비별 `bearer_token`을 복사할 필요가 없습니다.

인터넷 또는 신뢰하지 않는 망에서 Server 7070에 접근할 수 있다면 관리자가 enrollment token 보호 모드를 사용할 수 있습니다. 이 경우에만 다음 항목을 추가하고 Token 파일을 `root:invenqor-agent`, `0640`으로 설치합니다.

```
sudo install -m 0640 -o root -g invenqor-agent \
  enrollment.token /etc/invenqor-agent/enrollment.token
```

```
enrollment_token_file = "/etc/invenqor-agent/enrollment.token"
```

수동 등록이 필요한 예외 장비만 `bearer_token = "ivq_at_..."`을 사용합니다.

5.2 mTLS와 사설 CA 예시

```
[server]
url = "https://inventory.example.internal"
ca_file = "/etc/invenqor-agent/ca.pem"
client_identity_pem = "/etc/invenqor-agent/device.pem"
timeout_seconds = 30
```

`device.pem`은 클라이언트 인증서 체인과 개인키가 하나의 PEM에 있어야 합니다. 권한을 제한합니다.

```
sudo chown root:invenqor-agent \
  /etc/invenqor-agent/ca.pem /etc/invenqor-agent/device.pem
sudo chmod 0640 /etc/invenqor-agent/ca.pem
sudo chmod 0640 /etc/invenqor-agent/device.pem
```

5.3 설정 검증과 재시작

```
sudo -u invenqor-agent \
  /opt/invenqor-agent/bin/invenqor-agent \
  --config /etc/invenqor-agent/config.toml \
  --validate-config
```

`configuration is valid`가 출력되면 서비스를 재시작합니다.

```
sudo systemctl restart invenqor-agent
sudo systemctl status invenqor-agent --no-pager
```

OpenRC는 `sudo rc-service invenqor-agent restart`, SysV는 `sudo service invenqor-agent restart`를 사용합니다.

6. 정상 동작 확인

6.1 로그 확인

systemd:

```
sudo journalctl -u invenqor-agent --since "10 minutes ago" --no-pager
```

정상 수집 시 `queued collection event`, 정상 전송 시 `delivered queued events` 메시지를 확인할 수 있습니다. 서버 URL이 없으면 전송 메시지가 없는 것이 정상입니다.

실시간 로그:

```
sudo journalctl -u invenqor-agent -f
```

6.2 로컬 상태 확인

```
sudo ls -la /var/lib/invenqor-agent
sudo find /var/lib/invenqor-agent/queue -maxdepth 1 \
  -type f -name '*.jsonl' -printf '%f %s bytes\n'
sudo du -sh /var/lib/invenqor-agent/queue
```

파일·디렉터리	의미
`agent-id`	설치 단위를 식별하는 UUID, 권한 `0600`

파일·디렉터리	의미
`inventory.json`	직전 유효 인벤토리
`snapshot.sha256`	변경 감지용 안정 해시
`last-heartbeat`	마지막 이벤트/하트비트 시각
`queue/*.jsonl`	아직 서버가 수락하지 않은 이벤트

큐 파일은 서버가 2xx 응답과 `accepted: true`를 모두 반환한 후에만 삭제됩니다. 전송 장애 중 큐 파일이 증가하는 것은 데이터 보존 동작입니다.

6.3 한 번만 수집하기

`--once`는 수집 결과 JSON을 화면에 출력하고 전송도 한 번 시도합니다. 운영 서비스와 같은 상태 디렉터리를 동시에 사용하면 안 되므로 먼저 서비스를 멈춥니다.

```
sudo systemctl stop invenqor-agent
sudo -u invenqor-agent \
  /opt/invenqor-agent/bin/invenqor-agent \
  --config /etc/invenqor-agent/config.toml \
  --once
sudo systemctl start invenqor-agent
```

> **출력 취급:** 한 번 수집 결과에는 계정명, 설치 소프트웨어, 네트워크 주소, 프로세스 정보가 들어 있습니다. 티켓이나 채팅에 원문을 붙이지 말고 조직의 자산정보 등급에 따라 보관하십시오.

7. 일상 운영

7.1 기본 주기

- 전체 수집: 900초(15분)
- 변경이 없을 때 하트비트: 300초(5분)
- 전송 실패 재시도: 1초부터 2배씩 증가, 최대 3600초
- 로컬 큐 한도: 100 MiB
- 요청 타임아웃: 30초

설정 변경은 관리자 승인 아래 수행하십시오. 지나치게 짧은 주기는 서버와 게이트웨이 부하를 높이고, 지나치게 긴 주기는 자산 변경 탐지 시간을 늦춥니다.

7.2 서비스 명령

작업	systemd 명령
상태	`sudo systemctl status invenqor-agent`
시작	`sudo systemctl start invenqor-agent`
중지	`sudo systemctl stop invenqor-agent`
재시작	`sudo systemctl restart invenqor-agent`
부팅 시 자동 시작 확인	`sudo systemctl is-enabled invenqor-agent`
최근 로그	`sudo journalctl -u invenqor-agent -n 100`

7.3 프로세스 명령행 수집

기본값 `include_process_cmdline = false`를 유지하는 것을 권장합니다. 반드시 필요하다면 보안·개인정보 책임자의 승인, 중앙 저장소 접근 통제, 보존 기간을 먼저 확정한 뒤 활성화하십시오.

```
[collectors]
include_process_cmdline = true
```

명령행에는 데이터베이스 비밀번호, API 토큰, 개인 경로가 포함될 수 있습니다.

8. 문제 해결

8.1 서비스가 시작되지 않음

```
sudo systemctl status invenqor-agent --no-pager
sudo journalctl -u invenqor-agent -n 100 --no-pager
sudo -u invenqor-agent \
  /opt/invenqor-agent/bin/invenqor-agent \
  --config /etc/invenqor-agent/config.toml \
  --validate-config
```

주요 원인:

- TOML 오타 또는 지원하지 않는 설정 키
- 설정, CA, mTLS PEM 파일 읽기 권한 부족
- 상태 디렉터리 쓰기 권한 부족
- `interval_seconds`, `heartbeat_seconds`, `timeout_seconds`, `max_processes`가 0
- release 바이너리에 `http://` URL 사용

8.2 TLS 또는 인증 오류

확인 순서:

1. URL이 `https://` 로 시작하고 DNS가 올바른지 확인합니다.
2. 서버 시간이 크게 어긋나지 않았는지 확인합니다.
3. 사설 CA라면 `ca_file` 경로와 PEM 내용을 확인합니다.
4. mTLS라면 인증서 체인과 개인키가 같은 `client_identity_pem` 에 있는지, 만료되지 않았는지 확인합니다.
5. bearer token이 장비에 맞게 발급됐고 공백 없이 입력됐는지 확인합니다.
6. 게이트웨이가 `{"accepted":true}` JSON을 반환하는지 관리자에게 확인합니다.

Server가 요청을 거부하면 Agent 로그에는 다음처럼 HTTP 상태 외에 안전한 오류 코드, API 경로와 request ID가 함께 표시됩니다.

```
automatic enrollment returned HTTP 403 Forbidden
code=AGENT_SOURCE_NOT_ALLOWED path=/v1/agent/enroll
request_id=01J... message=The Agent source IP is not permitted...
```

`request_id` 전체를 관리자에게 전달하십시오. 관리자는 **Server 로그** 화면에서 같은 ID를 검색해 요청을 처리한 Pod, 판정 IP, 정책 버전과 실패 단계를 확인할 수 있습니다. Agent 로그나 문의 내용에 등록 Token, 장비 Token, Secret, 원문 인벤토리를 붙이지 마십시오.

8.3 수집기 오류

에이전트는 한 수집기가 실패해도 나머지 수집을 계속합니다. 권한, 오래된 배포판의 명령 차이, `/proc` 또는 `/sys` 마운트 제한을 확인하십시오. 수집기 오류가 있는 주기에는 누락된 자산을 삭제로 판단하지 않아 잘못된 대량 삭제를 방지합니다.

8.4 큐가 계속 증가함

```
sudo du -sh /var/lib/invenqor-agent/queue
sudo find /var/lib/invenqor-agent/queue -type f -name '*.jsonl' | wc -l
```

게이트웨이 연결, TLS, 인증, 응답 형식을 확인합니다. 큐 파일을 수동 삭제하면 미전송 인벤토리를 잃습니다. 복구 전 삭제하지 말고 관리자에게 전달하십시오. 큐가 한도에 도달하면 기존 이벤트를 보존하고 새 이벤트 생성을 실패시킵니다.

9. 제거

압축을 풀어 둔 원본 패키지 디렉터리에서 실행합니다.

```
sudo ./scripts/uninstall.sh
```

제거 스크립트는 서비스와 바이너리를 제거하지만 다음 데이터는 의도적으로 보존합니다.

- `/etc/invenqor-agent` : 설정, 인증서 참조 파일

- `/var/lib/invenqor-agent` : 장비 ID, 직전 인벤토리, 미전송 큐

보존 데이터까지 삭제하려면 먼저 감사·복구·보존 정책과 미전송 여부를 확인한 후 관리자가 별도로 처리해야 합니다.

10. 자주 묻는 질문

서버에 inbound 포트를 열어야 합니까?

아닙니다. 에이전트는 설정된 게이트웨이로 HTTPS outbound 연결만 시작합니다.

게이트웨이 없이 사용할 수 있습니까?

가능합니다. `server.url`을 생략하면 로컬 수집과 큐 보존만 수행합니다. 장기간 운영하면 큐 용량 정책이 필요합니다.

수집 실패가 에이전트 전체 중단으로 이어집니까?

대부분의 수집기는 서로 격리되어 있어 한 수집기의 오류가 나머지를 중단시키지 않습니다. 다만 상태 저장 실패나 잘못된 설정처럼 핵심 기능 오류는 실행에 영향을 줄 수 있습니다.

에이전트가 취약점을 판정합니까?

아닙니다. 패키지 인벤토리는 수집하지만 CVE 매핑, 위험도 계산, 정책 판정은 중앙 시스템의 역할입니다.

자동 업데이트나 원격 명령을 지원합니까?

서명된 Agent 자동 업데이트는 선택적으로 지원합니다. 관리자가 승인·서명한 새 버전만 다운로드하고 SHA-256, 크기, OS/Architecture와 고정 Ed25519 공개키를 검증한 뒤 스테이징합니다. systemd 환경에서는 전용 root helper가 원자 교체하고 이전 바이너리를 `.previous`로 보존합니다. 설정 방법은 Server 설치 및 운영 가이드를 참조하십시오. 원격 셸이나 임의 명령 실행은 지원하지 않습니다.

11. 관리 콘솔 로그인과 계정

로그인 화면 하단에는 현재 Server 버전이 표시됩니다. 로그인 후에는 상단 버전 chip에서 같은 버전과 build 정보를 확인할 수 있어 장애 문의 시 실행 버전을 정확히 전달할 수 있습니다.

조직 관리자가 Keycloak을 활성화한 경우에만 **Keycloak으로 계속** 버튼이 나타납니다. 버튼을 누르면 조직 로그인 화면으로 이동하며, 인증 후 원래 Invenqor 서비스로 돌아옵니다. Keycloak 계정의 이름, Email과 SSO 역할은 로그인 때마다 조직 정책에 맞게 동기화됩니다.

- 로컬 계정 비밀번호를 잊은 경우 Invenqor 관리자에게 초기화를 요청합니다.
- Keycloak 비밀번호·MFA·잠금은 조직 Keycloak 관리자에게 문의합니다.
- 계정이 비활성화되면 기존 브라우저 Session과 API key도 사용할 수 없습니다.
- 화면 메뉴는 부여된 역할의 권한에 따라 자동으로 숨겨집니다.
- SSO 장애 시 로컬 비상 관리자 계정 사용 여부는 조직 운영 절차를 따릅니다.

12. 관리 콘솔 사용

운영 현황은 Server가 PostgreSQL에서 실시간 집계한 값만 사용합니다. 관리 자산 수, 최근 24시간 내 확인된 자산, 30분 내 연결된 Agent, 최근 24시간 수집 이벤트와 실패 건수를 KPI로 표시합니다. 7일 수집 추이, 중요도·환경·유형·수집 원천 분포와 점검 필요 자산/Agent를 함께 보고 수집 공백을 우선 처리하십시오.

주요 화면과 실제 동작은 다음과 같습니다.

화면	사용할 수 있는 기능
자산	검색·필터·페이징, 등록·수정·삭제·복원, 상세·원천·이력·관계 조회
자산 고급 작업	선택 자산 병합, 선택 수집 원천을 새 자산으로 분리, 관계 생성·삭제
Agent	상태·최근 수집 시각 확인, 예외 장비 수동 등록, Token 회전, 차단·해제, mTLS 인증서 등록
설정 → Agent 등록	신규 Agent의 URL-only/Token 모드, IP/CIDR allowlist와 신뢰 프록시를 즉시 전환
Query DSL	실행 전 구문과 AST 검증, 제한 건수 내 결과 조회
API · MCP 키	최소권한 scope 선택, 이름 변경, scope 추가·삭제, 무중단 회전과 폐기
감사 로그	행위자·대상·요청 ID·IP·사유와 변경 전후 데이터 확인
Server 로그	모든 Pod의 Agent 등록·전송 실패와 Server 오류를 request ID로 검색
내 보안	로컬 비밀번호 변경, TOTP 설정·활성화·비활성화
우측 상단 프로필	내 보안, 개인화, 로그아웃으로 이동
개인화	테마, 화면 밀도, 시작 화면, 통계 갱신 주기, 모션 축소

화면의 생성·수정·삭제 버튼은 해당 권한이 있을 때만 나타납니다. 작업 실패 시 표시되는 서버 오류를 확인하고, 권한 오류는 관리자에게 역할을 요청하십시오. 페이지를 새로 열거나 Keycloak에서 돌아온 경우에도 브라우저는 현재 Session에 연결된 CSRF 보호 값을 자동 사용하므로 사용자가 Token을 복사할 필요가 없습니다. 개인화 설정은 사용자 ID별로 현재 브라우저에 저장되며 다른 사용자나 Server 운영 설정에는 영향을 주지 않습니다. 현재 주 메뉴와 **설정**의 하위 메뉴는 `#/settings/keycloak` 같은 URL과 사용자별 브라우저 상태에 함께 저장되어 새로고침·뒤로가기·다시 로그인 후에도 유지됩니다. 권한이 회수된 메뉴나 잘못된 URL은 접근 가능한 첫 화면으로 안전하게 복구됩니다.