SECURITY ANALYZER GENERAL STUDY

BY IDEO-LAB

VERSION 1

AUTHOR : GUILLAUME ONEILL

SECURITY ANALYZER GUIDE

IDEO-LAB

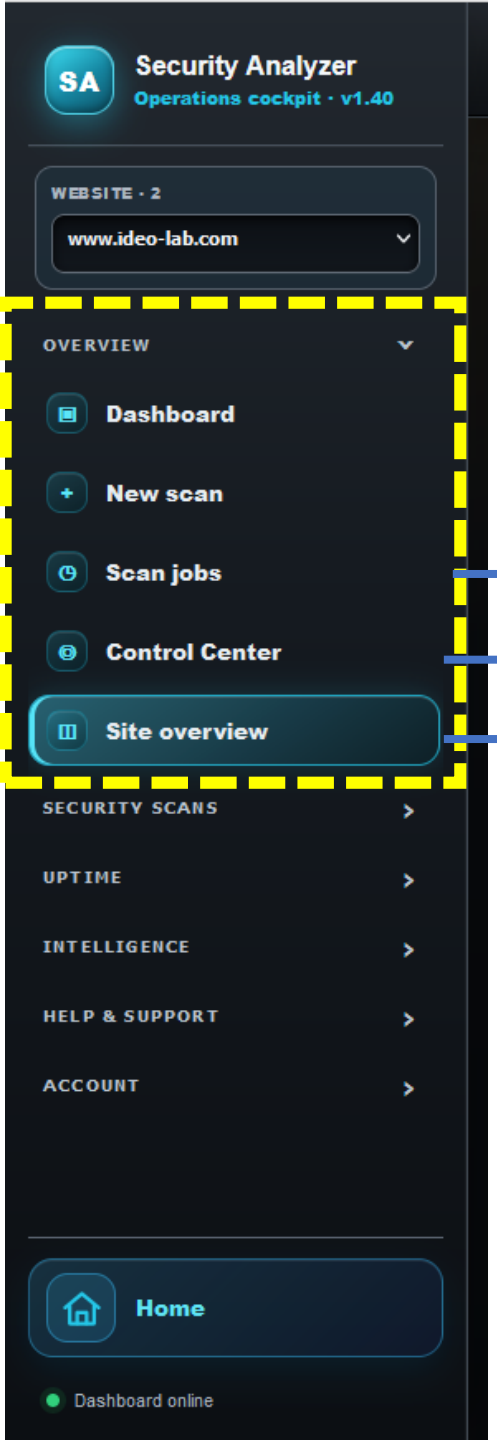
AUGUST 2026

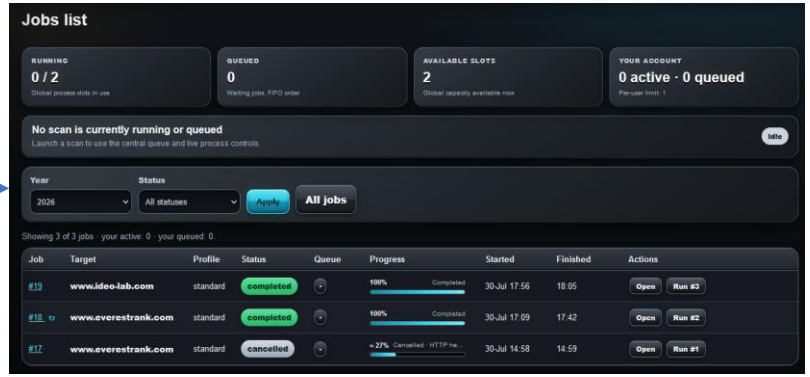
Version : 1.46

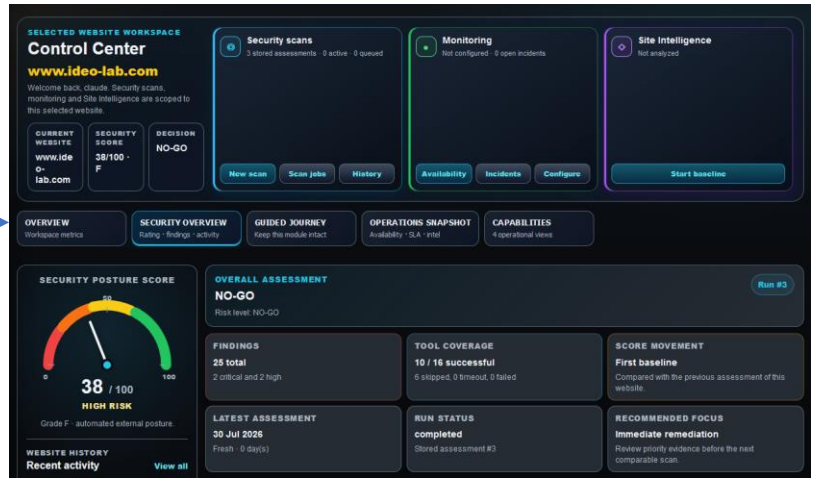
AGENDA

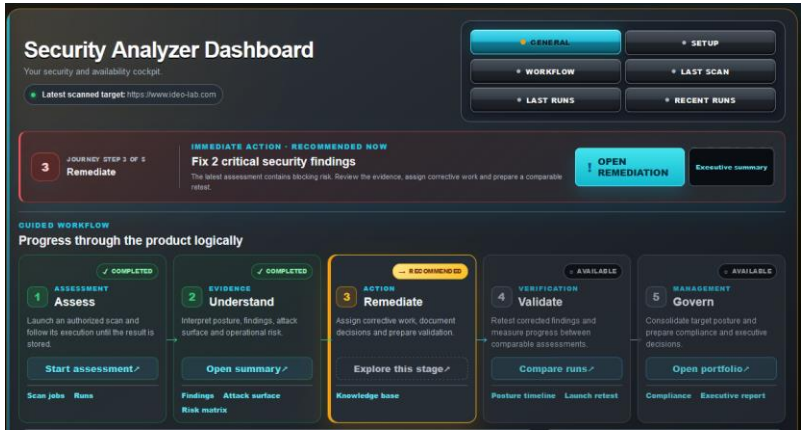
- 1. Signup
- 2. Login
- 3. Scan Web site
- 4. Site Overview

0. Side Menu


SA
Security Analyzer
Operations cockpit · v1.40
WEBSITE · 2
www.ideo-lab.com
OVERVIEW
Dashboard
New scan
Scan jobs
Control Center
Site overview
SECURITY SCANS
UPTIME
INTELLIGENCE
HELP & SUPPORT
ACCOUNT
Home
Dashboard online


Jobs list
RUNNING
0 / 2
QUEUED
0
AVAILABLE SLOTS
2
YOUR ACCOUNT
0 active · 0 queued
No scan is currently running or queued
Launch a scan to use the central queue and live process controls.
idle
Year
Status
2026
All statuses
Apply
All jobs
Showing 3 of 3 jobs · your active: 0 · your queued: 0.
Job
Target
Profile
Status
Queue
Progress
Started
Finished
Actions
#19
www.ideo-lab.com
standard
completed
30-Jul 17:56
18:05
Open
Run #3
#18
www.everestrank.com
standard
completed
30-Jul 17:09
17:42
Open
Run #2
#17
www.everestrank.com
standard
cancelled
30-Jul 14:58
14:59
Open
Run #1


SELECTED WEBSITE WORKSPACE
Control Center
www.ideo-lab.com
Welcome back, claude. Security scans, monitoring and Site Intelligence are scoped to this selected website.
CURRENT WEBSITE
www.ideo-lab.com
SECURITY SCORE
38/100 · F
DECISION
NO-GO
Security scans
3 stored assessments · 0 active · 0 queued
New scan
Scan jobs
History
Monitoring
Not configured · 0 open incidents
Availability
Incidents
Configure
Site intelligence
Not analyzed
Start baseline
OVERVIEW
Workspace metrics
SECURITY OVERVIEW
Rating · findings · activity
GUIDED JOURNEY
Keep this module intact
OPERATIONS SNAPSHOT
Availability · SLA · intel
CAPABILITIES
4 operational views
SECURITY POSTURE SCORE
38 / 100
HIGH RISK
Grade F · automated external posture.
WEBSITE HISTORY
Recent activity
View all
OVERALL ASSESSMENT
NO-GO
Risk level: NO-GO
Run #3
FINDINGS
25 total
2 critical and 2 high
TOOL COVERAGE
10 / 16 successful
6 skipped, 0 timeout, 0 failed
SCORE MOVEMENT
First baseline
Compared with the previous assessment of this website.
LATEST ASSESSMENT
30 Jul 2026
Fresh · 0 day(s)
RUN STATUS
completed
Stored assessment #3
RECOMMENDED FOCUS
Immediate remediation
Review priority evidence before the next comparable scan.


Security Analyzer Dashboard
Your security and availability cockpit.
Latest scanned target: https://www.ideo-lab.com
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT RUNS
JOURNEY STEP 3 OF 5
Remediate
IMMEDIATE ACTION · RECOMMENDED NOW
Fix 2 critical security findings
The latest assessment contains blocking risk. Review the evidence, assign corrective work and prepare a comparable retest.
OPEN REMEDIATION
Executive summary
GUIDED WORKFLOW
Progress through the product logically
COMPLETED
ASSESSMENT
1 Assess
Launch an authorized scan and follow its execution until the result is stored.
Start assessment
Scan jobs · Runs
COMPLETED
EVIDENCE
2 Understand
Interpret posture, findings, attack surface and operational risk.
Open summary
Findings · Attack surface · Risk matrix
RECOMMENDED
ACTION
3 Remediate
Assign corrective work, document decisions and prepare validation.
Explore this stage
Knowledge base
AVAILABLE
VERIFICATION
4 Validate
Retest corrected findings and measure progress between comparable assessments.
Compare runs
Posture timeline · Launch retest
AVAILABLE
MANAGEMENT
5 Govern
Consolidate target posture and prepare compliance and executive decisions.
Open portfolio
Compliance · Executive report

1. SIGNUP

SA

Security Analyzer
IDEO-Lab · version 1.40

Product | About | Authors | Team | Contact | Sign in | Create account

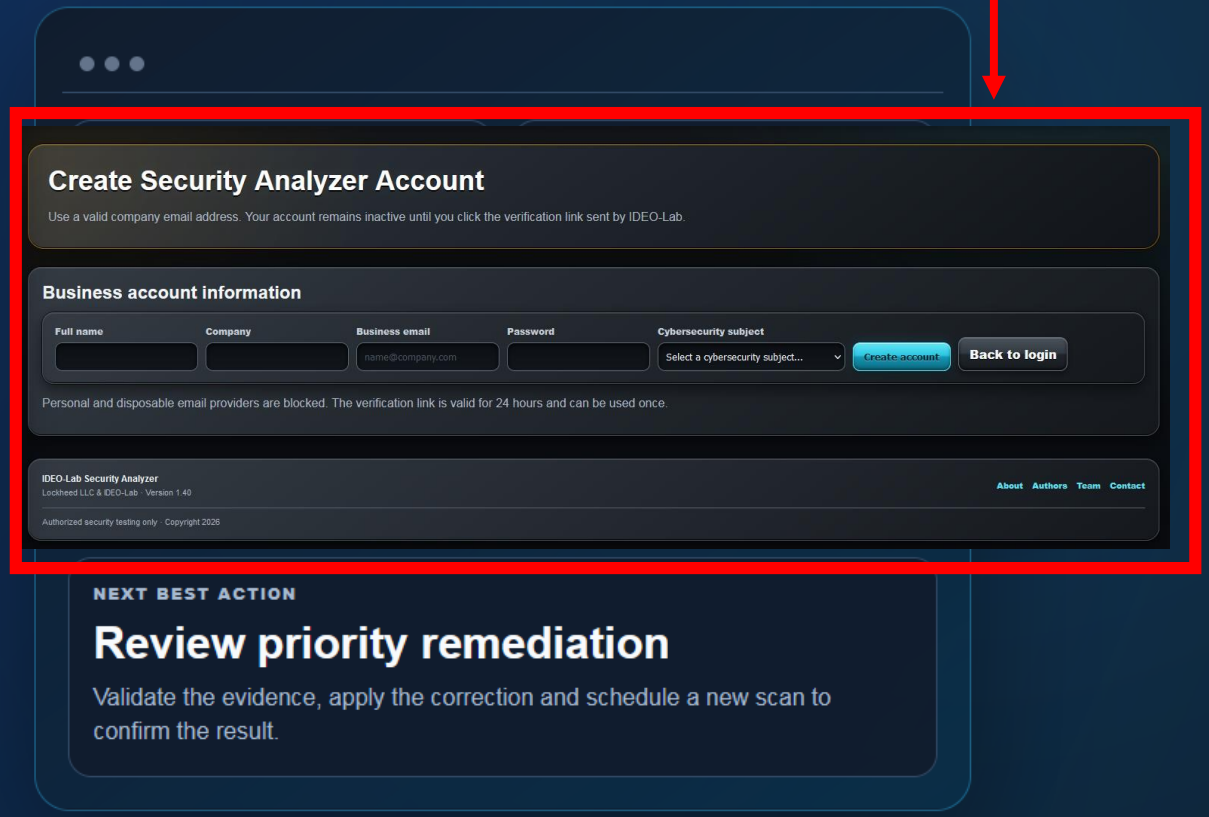
SECURITY · AVAILABILITY · INTELLIGENCE

Understand your exposure. Act on what matters.

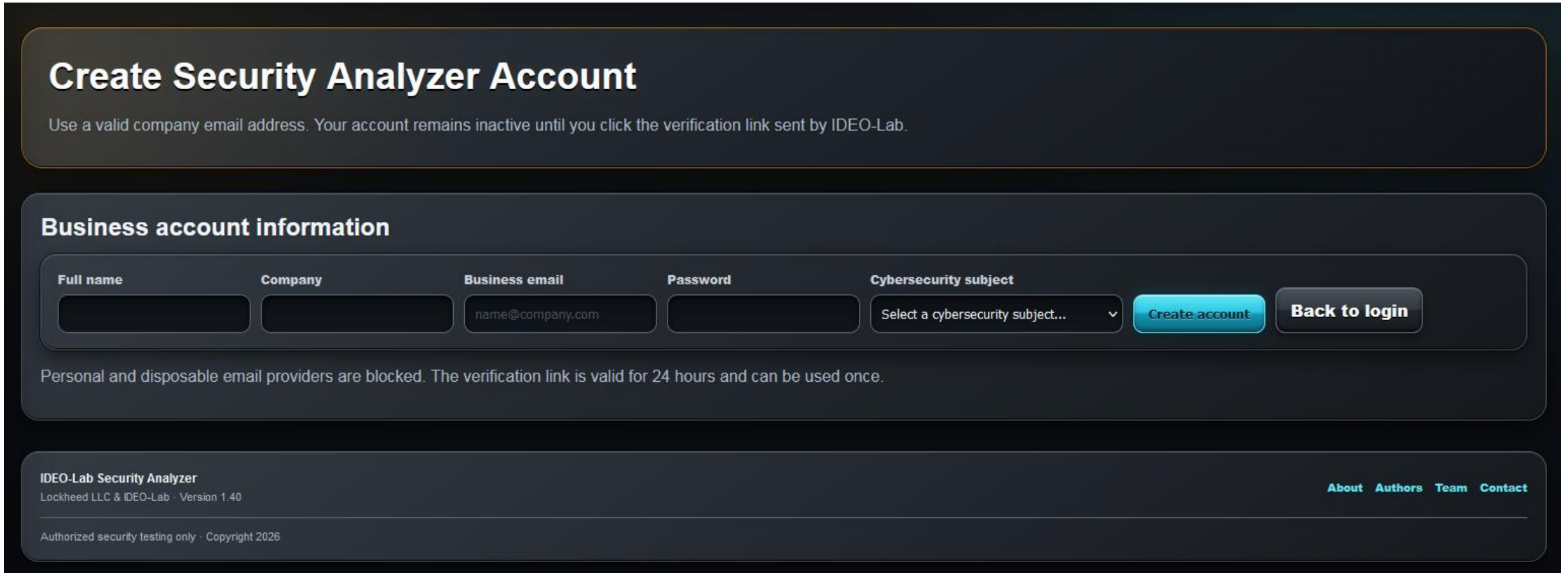
Security Analyzer combines coordinated security scanning, remediation workflows, uptime monitoring, SLA reporting and public website intelligence in one operational dashboard.

Create your workspace | **See the real interface** | **Sign in**

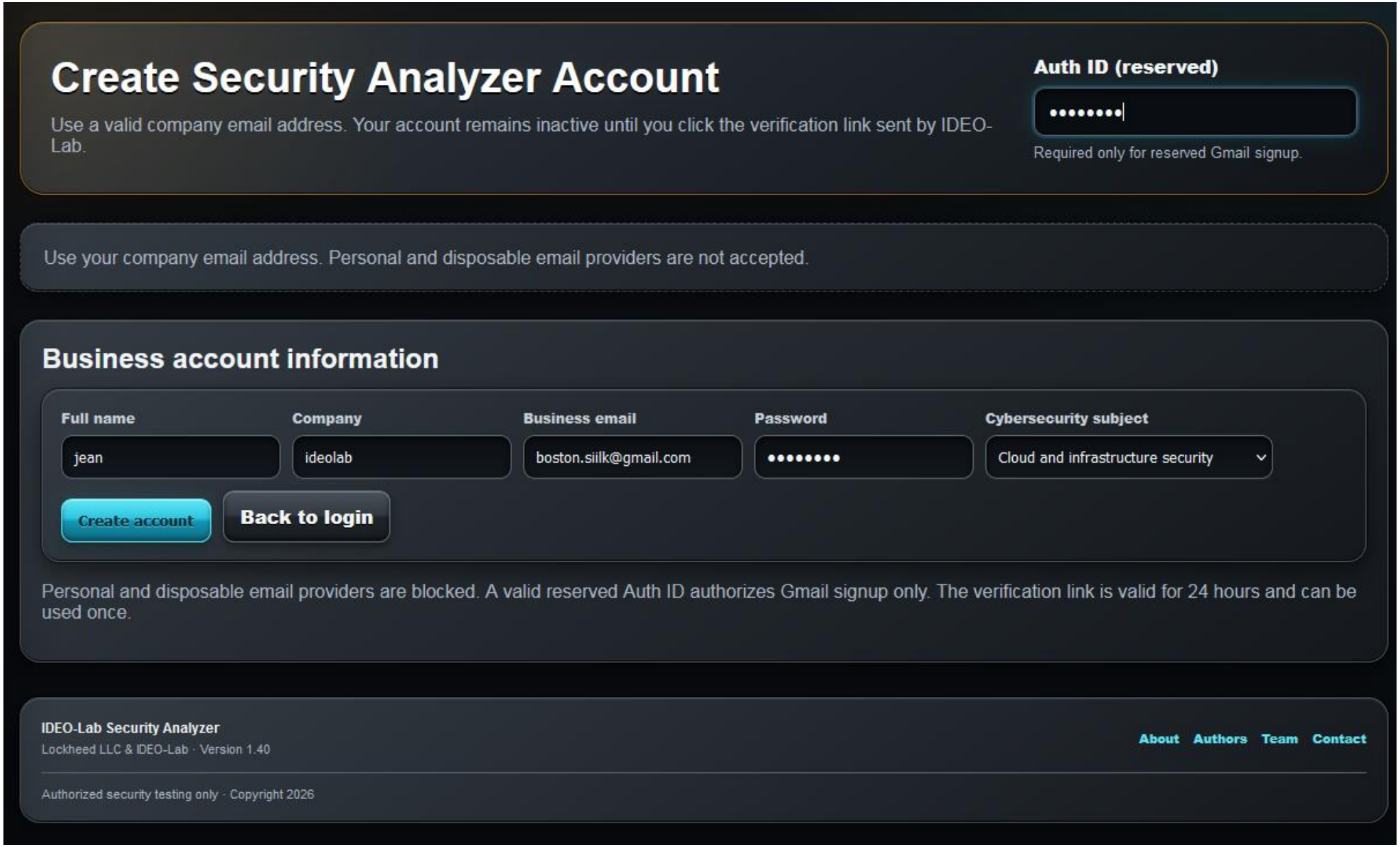
- Private account workspace
- Authorized targets only
- Centralized remediation



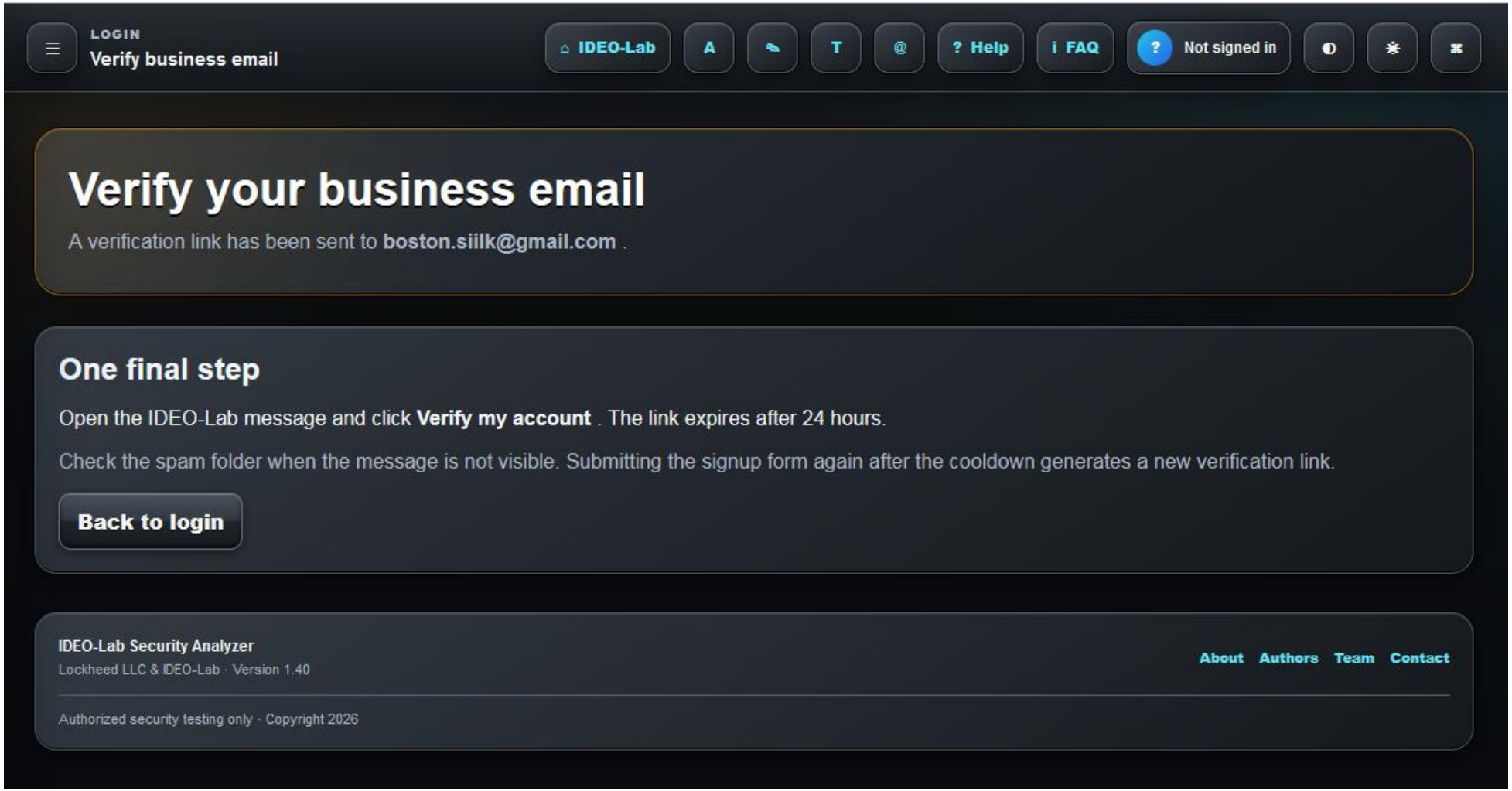
SIGNUP


Create Security Analyzer Account
Use a valid company email address. Your account remains inactive until you click the verification link sent by IDEO-Lab.
Business account information
Full name
Company
Business email
name@company.com
Password
Cybersecurity subject
Select a cybersecurity subject...
Create account
Back to login
Personal and disposable email providers are blocked. The verification link is valid for 24 hours and can be used once.
IDEO-Lab Security Analyzer
Lockheed LLC & IDEO-Lab · Version 1.40
About
Authors
Team
Contact
Authorized security testing only · Copyright 2026

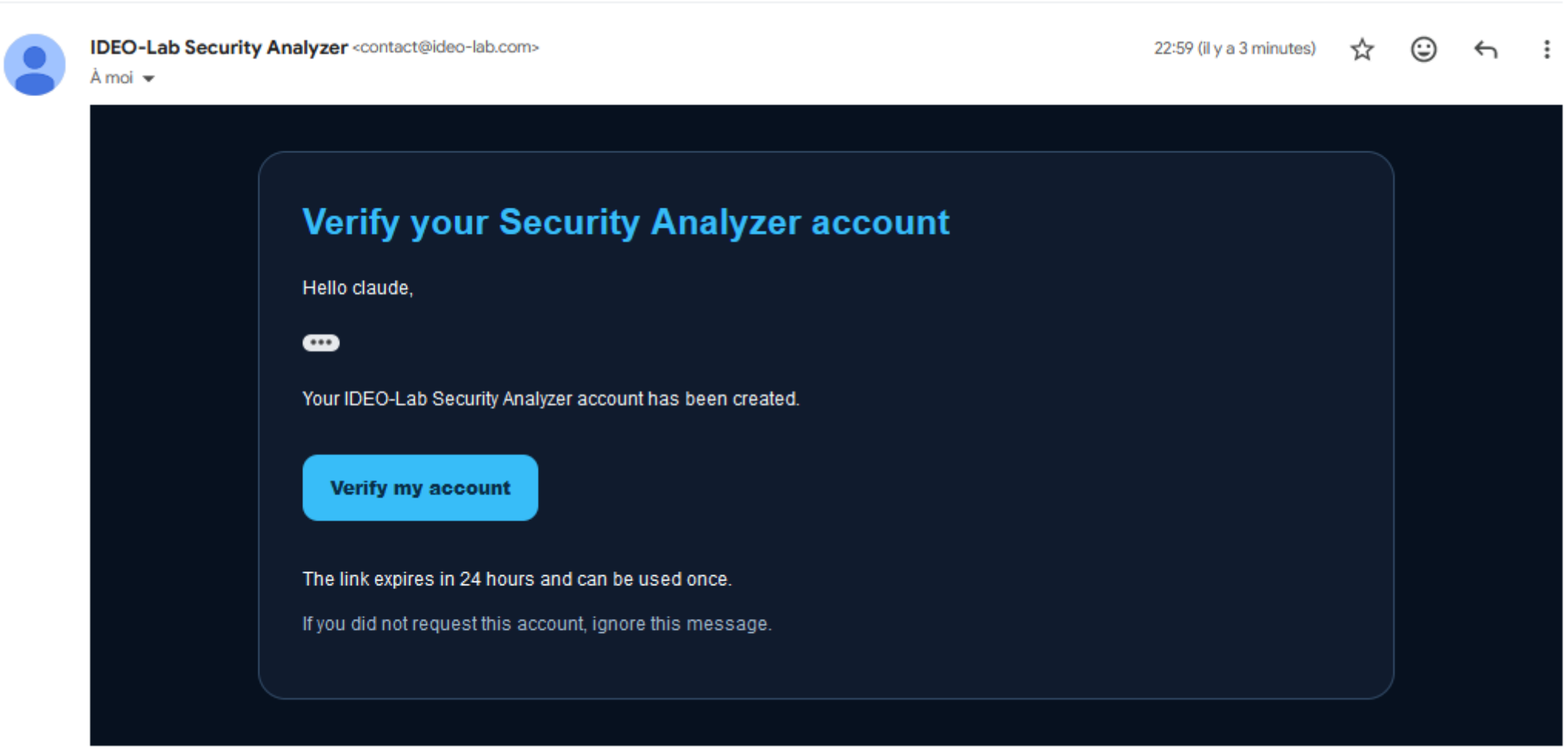
SIGNUP IGNITION



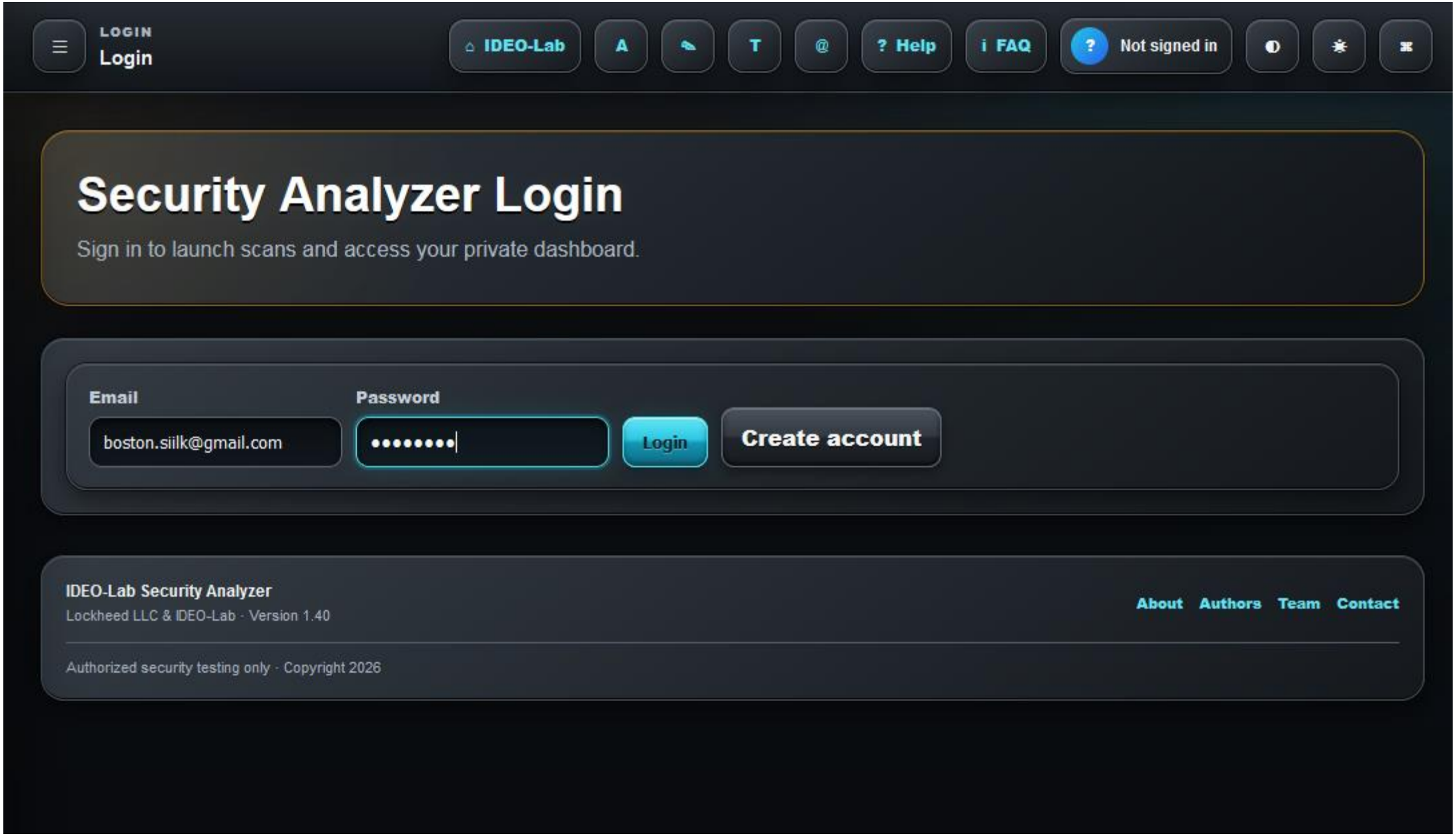
SEND EMAIL VERIFICATION



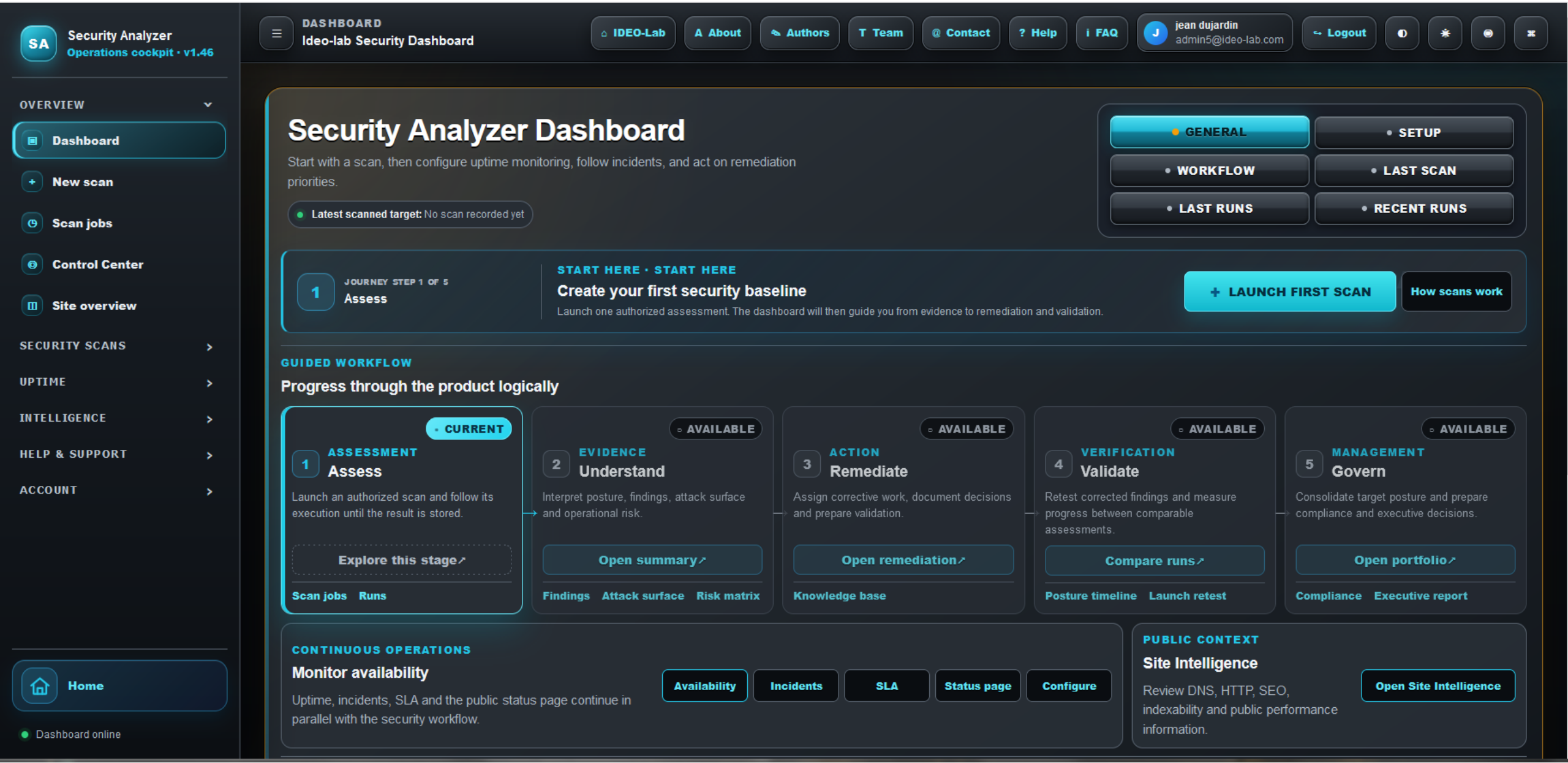
Email validated



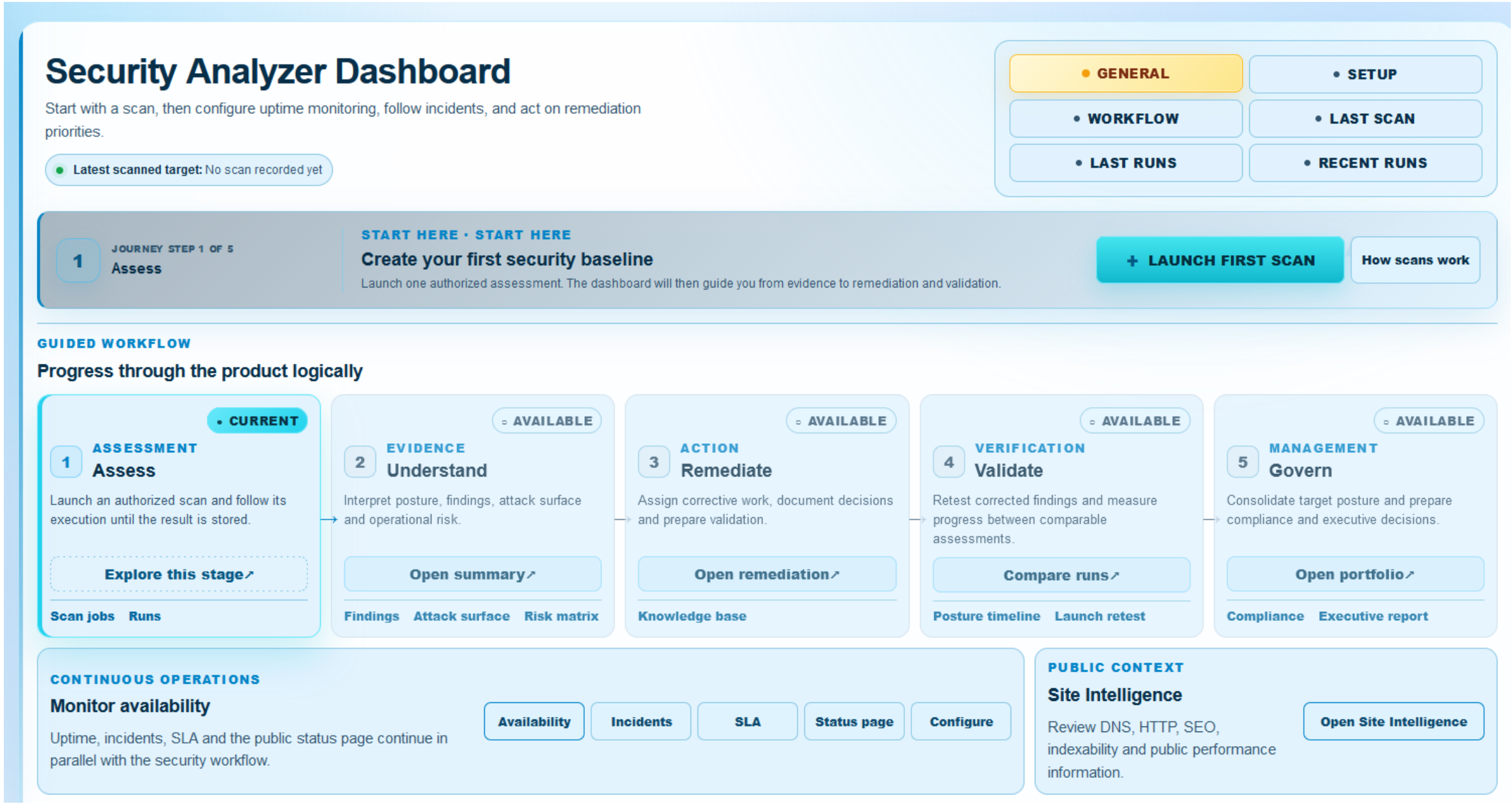
2. Login


LOGIN
Login
IDEO-Lab
A
T
@
Help
FAQ
Not signed in
Security Analyzer Login
Sign in to launch scans and access your private dashboard.
Email
boston.siilk@gmail.com
Password
Login
Create account
IDEO-Lab Security Analyzer
Lockheed LLC & IDEO-Lab · Version 1.40
About
Authors
Team
Contact
Authorized security testing only · Copyright 2026

First Login

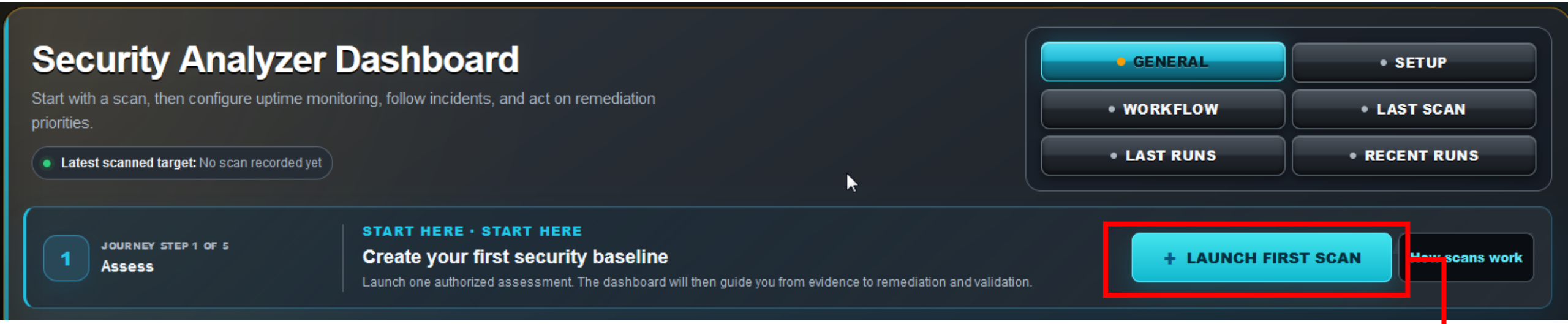

SA
Security Analyzer
Operations cockpit · v1.46
OVERVIEW
Dashboard
New scan
Scan jobs
Control Center
Site overview
SECURITY SCANS
UPTIME
INTELLIGENCE
HELP & SUPPORT
ACCOUNT
Home
Dashboard online
DASHBOARD
Ideo-lab Security Dashboard
IDEO-Lab
About
Authors
Team
Contact
Help
FAQ
jean dujardin
admin5@ideo-lab.com
Logout
Security Analyzer Dashboard
Start with a scan, then configure uptime monitoring, follow incidents, and act on remediation priorities.
Latest scanned target: No scan recorded yet
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT RUNS
1
JOURNEY STEP 1 OF 5
Assess
START HERE · START HERE
Create your first security baseline
Launch one authorized assessment. The dashboard will then guide you from evidence to remediation and validation.
LAUNCH FIRST SCAN
How scans work
GUIDED WORKFLOW
Progress through the product logically
CURRENT
1
ASSESSMENT
Assess
Launch an authorized scan and follow its execution until the result is stored.
Explore this stage
Scan jobs
Runs
AVAILABLE
2
EVIDENCE
Understand
Interpret posture, findings, attack surface and operational risk.
Open summary
Findings
Attack surface
Risk matrix
AVAILABLE
3
ACTION
Remediate
Assign corrective work, document decisions and prepare validation.
Open remediation
Knowledge base
AVAILABLE
4
VERIFICATION
Validate
Retest corrected findings and measure progress between comparable assessments.
Compare runs
Posture timeline
Launch retest
AVAILABLE
5
MANAGEMENT
Govern
Consolidate target posture and prepare compliance and executive decisions.
Open portfolio
Compliance
Executive report
CONTINUOUS OPERATIONS
Monitor availability
Uptime, incidents, SLA and the public status page continue in parallel with the security workflow.
Availability
Incidents
SLA
Status page
Configure
PUBLIC CONTEXT
Site Intelligence
Review DNS, HTTP, SEO, indexability and public performance information.
Open Site Intelligence

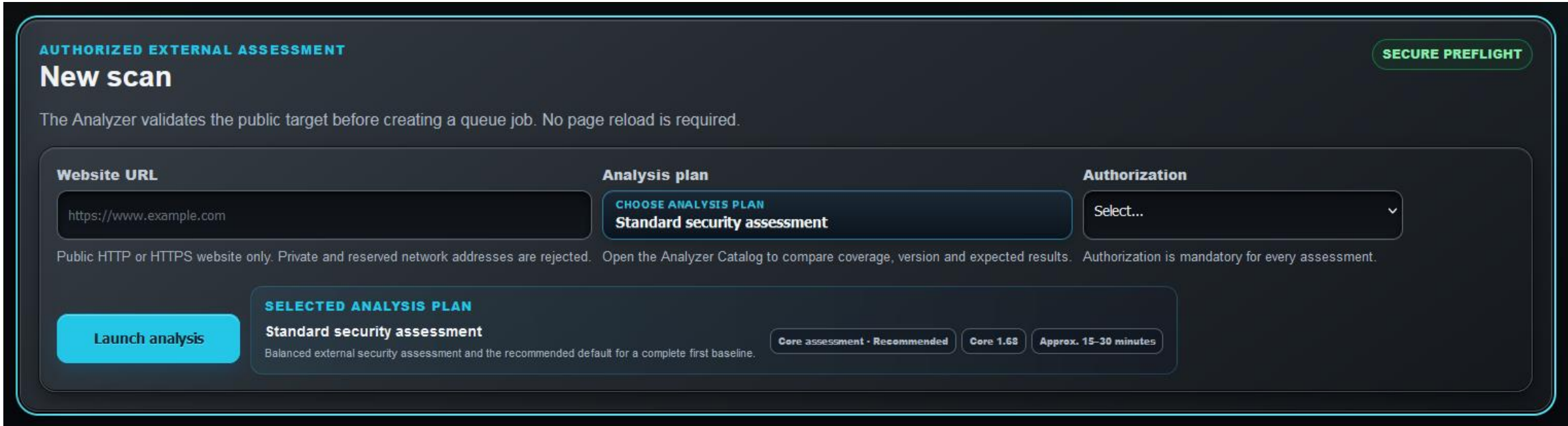
Light Theme


Security Analyzer Dashboard
Start with a scan, then configure uptime monitoring, follow incidents, and act on remediation priorities.
Latest scanned target: No scan recorded yet
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT RUNS
1
JOURNEY STEP 1 OF 5
Assess
START HERE · START HERE
Create your first security baseline
Launch one authorized assessment. The dashboard will then guide you from evidence to remediation and validation.
+ LAUNCH FIRST SCAN
How scans work
GUIDED WORKFLOW
Progress through the product logically
CURRENT
1
ASSESSMENT
Assess
Launch an authorized scan and follow its execution until the result is stored.
Explore this stage↗
Scan jobs
Runs
AVAILABLE
2
EVIDENCE
Understand
Interpret posture, findings, attack surface and operational risk.
Open summary↗
Findings
Attack surface
Risk matrix
AVAILABLE
3
ACTION
Remediate
Assign corrective work, document decisions and prepare validation.
Open remediation↗
Knowledge base
AVAILABLE
4
VERIFICATION
Validate
Retest corrected findings and measure progress between comparable assessments.
Compare runs↗
Posture timeline
Launch retest
AVAILABLE
5
MANAGEMENT
Govern
Consolidate target posture and prepare compliance and executive decisions.
Open portfolio↗
Compliance
Executive report
CONTINUOUS OPERATIONS
Monitor availability
Uptime, incidents, SLA and the public status page continue in parallel with the security workflow.
Availability
Incidents
SLA
Status page
Configure
PUBLIC CONTEXT
Site Intelligence
Review DNS, HTTP, SEO, indexability and public performance information.
Open Site Intelligence

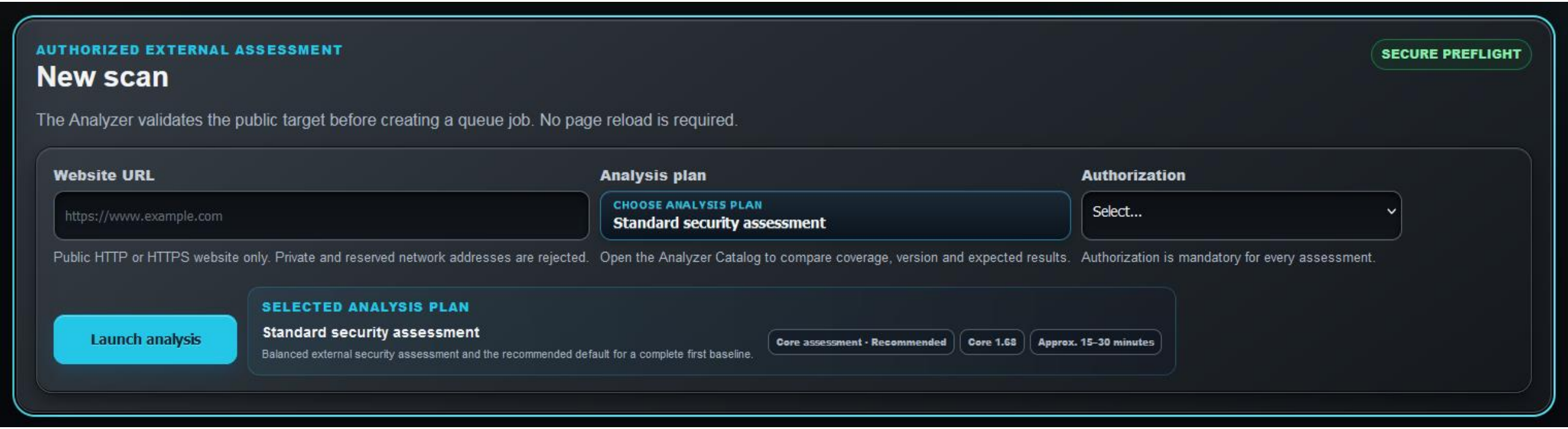
3. Scan Web site

Launch first scan

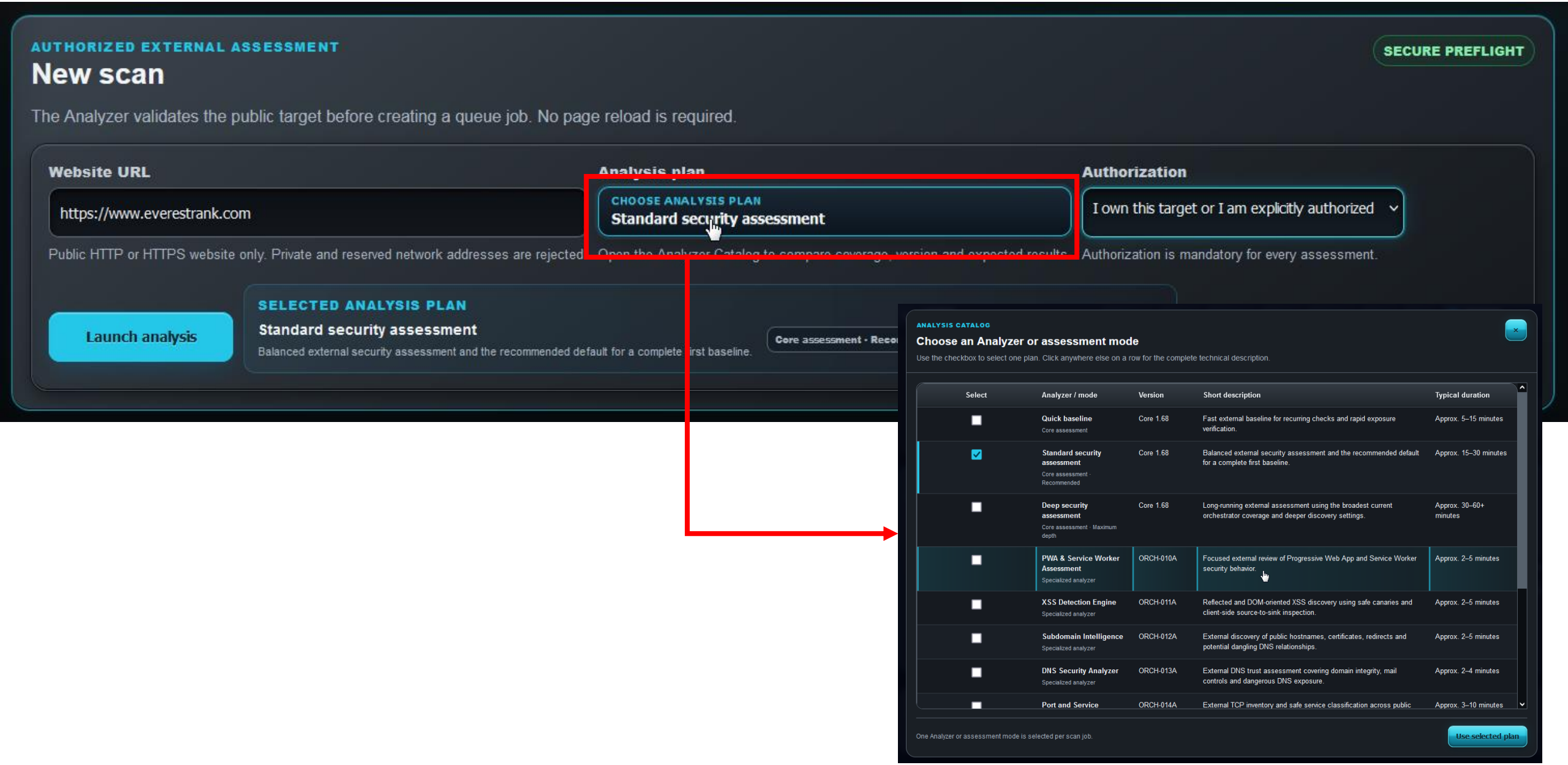




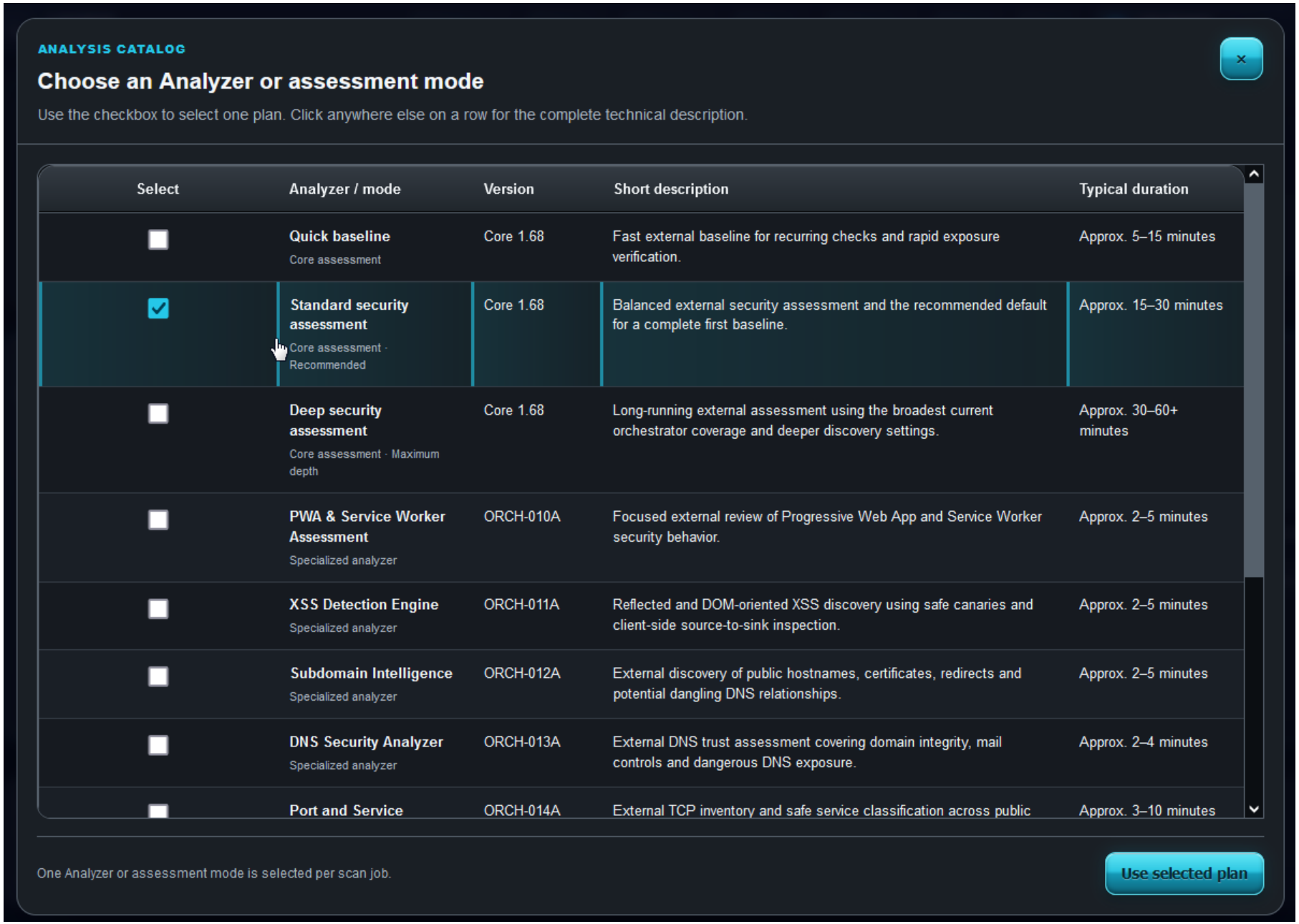
Start a New Web Scan



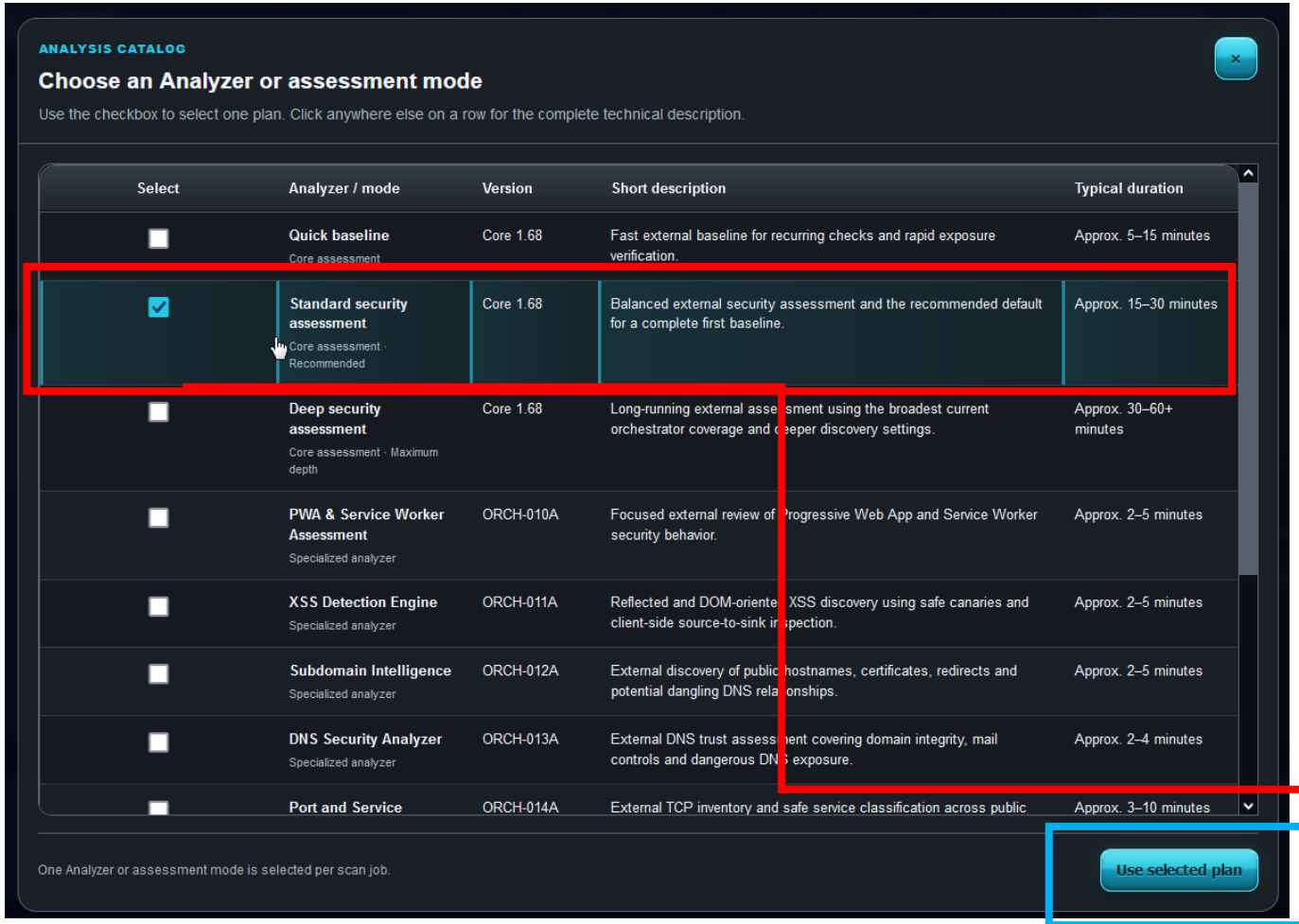
SELECT YOUR ANALYSIS PLAN



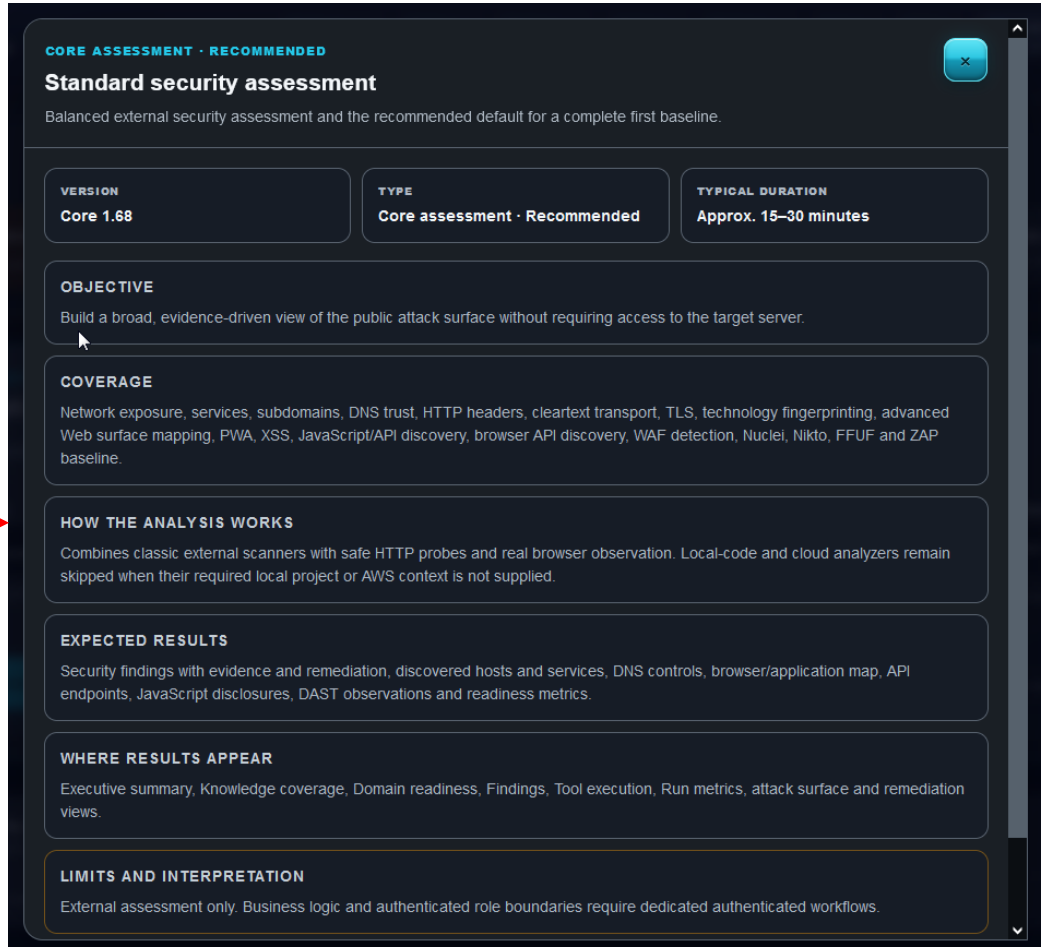
SELECT YOUR ANALYSIS PLAN

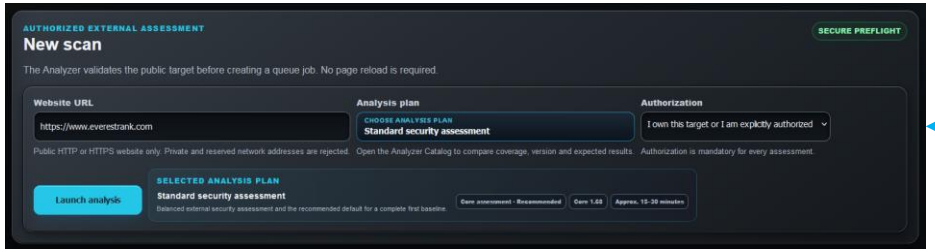


GET DETAILES INFORMATION ABOUT A PLAN



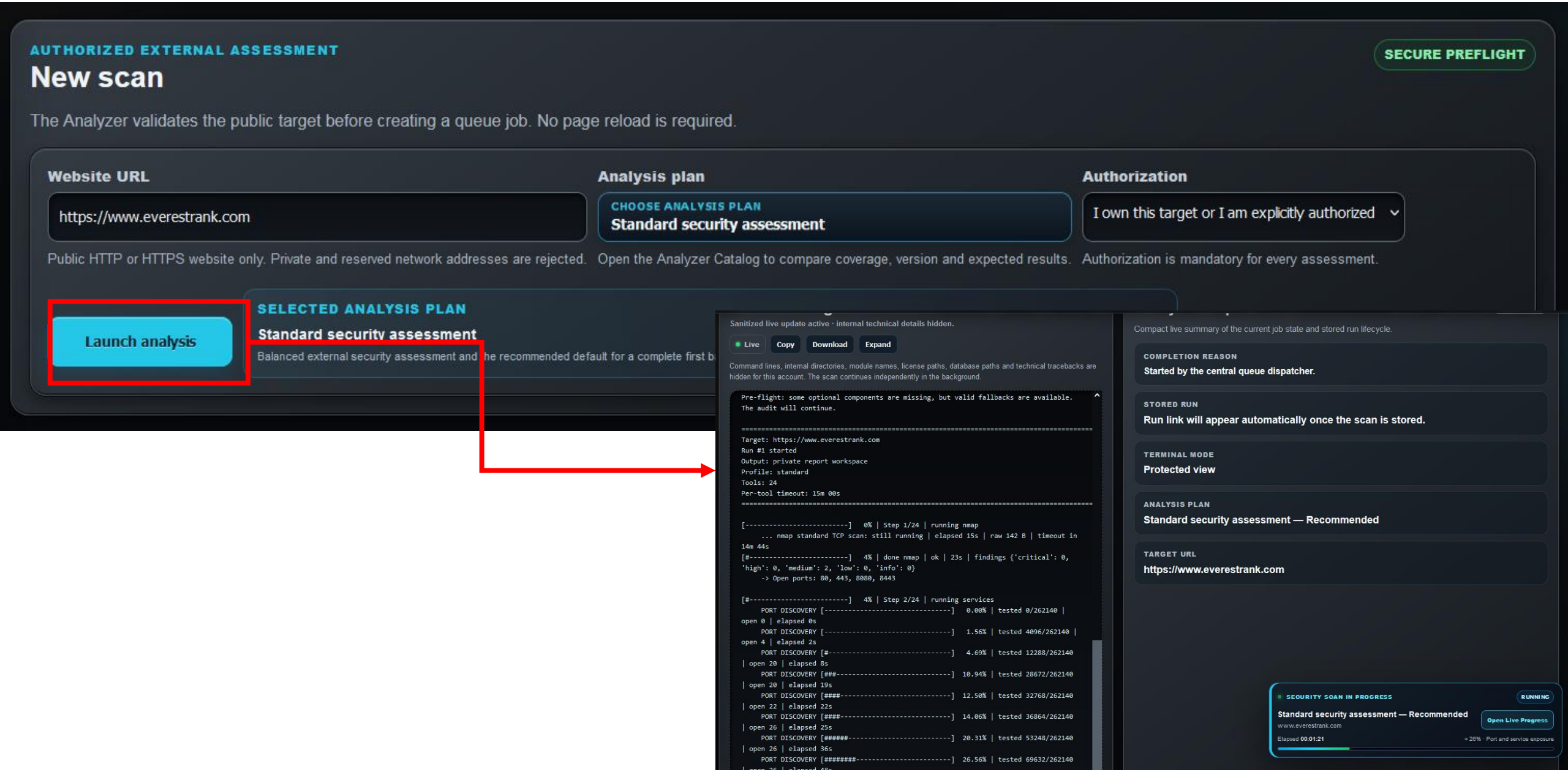
Click on a Plan to open An explanation Modal



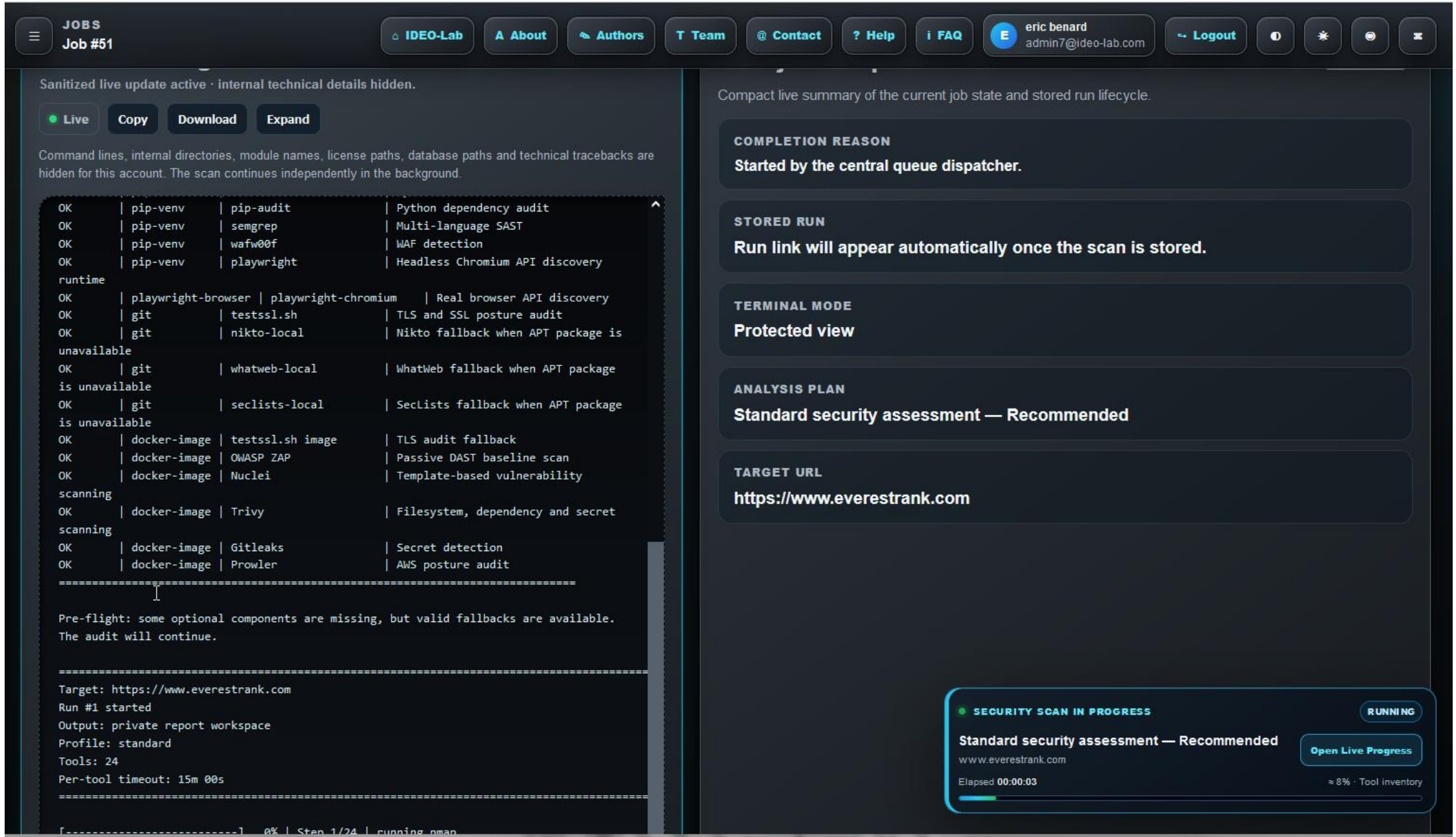


Click to validate & return

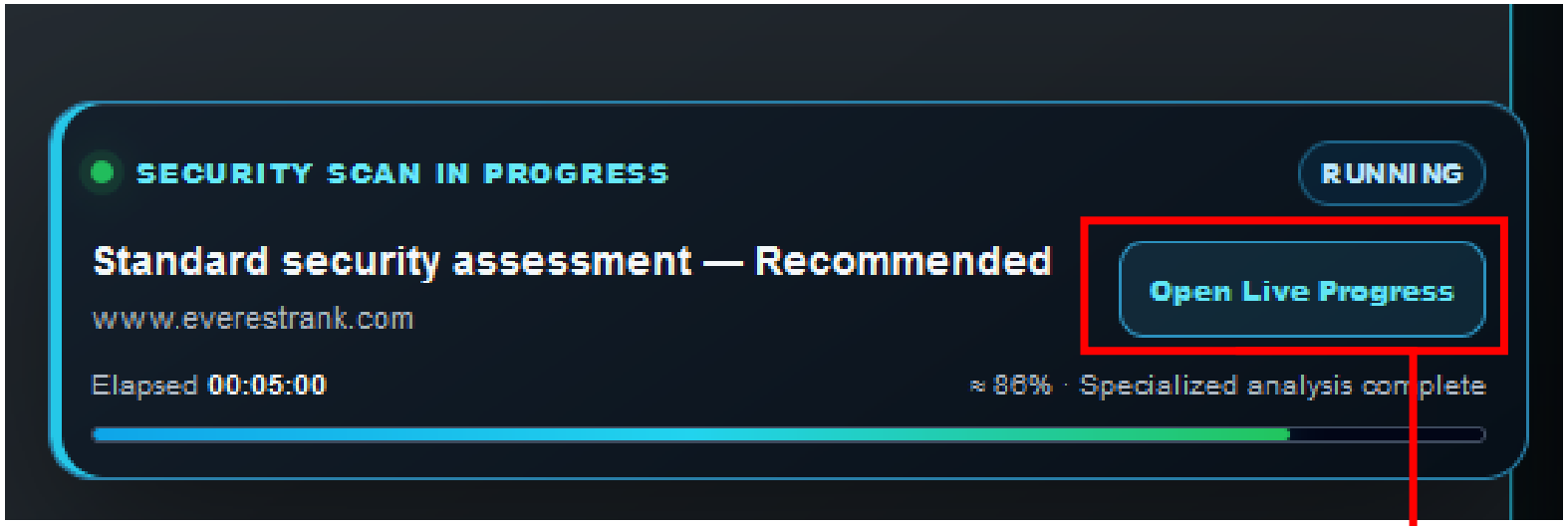
LAUNCH SCAN FORMULAR READY

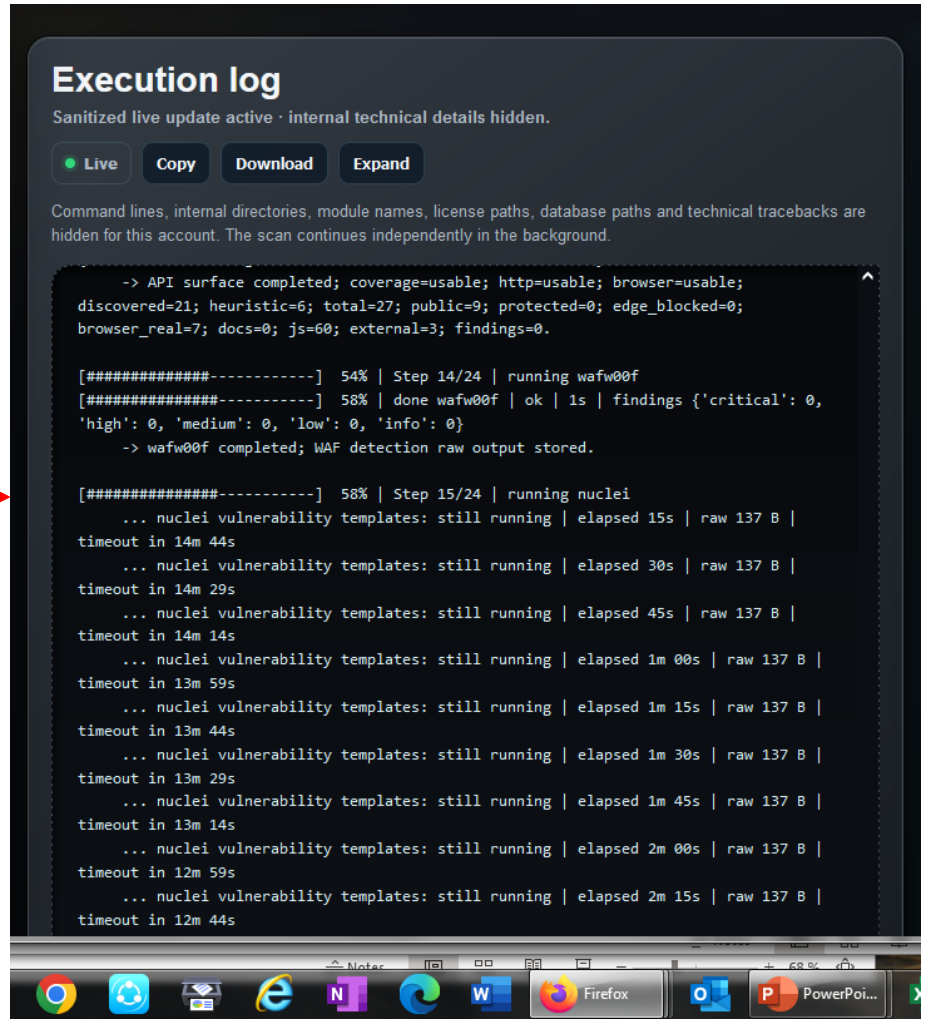


SCAN IN PROGRESS

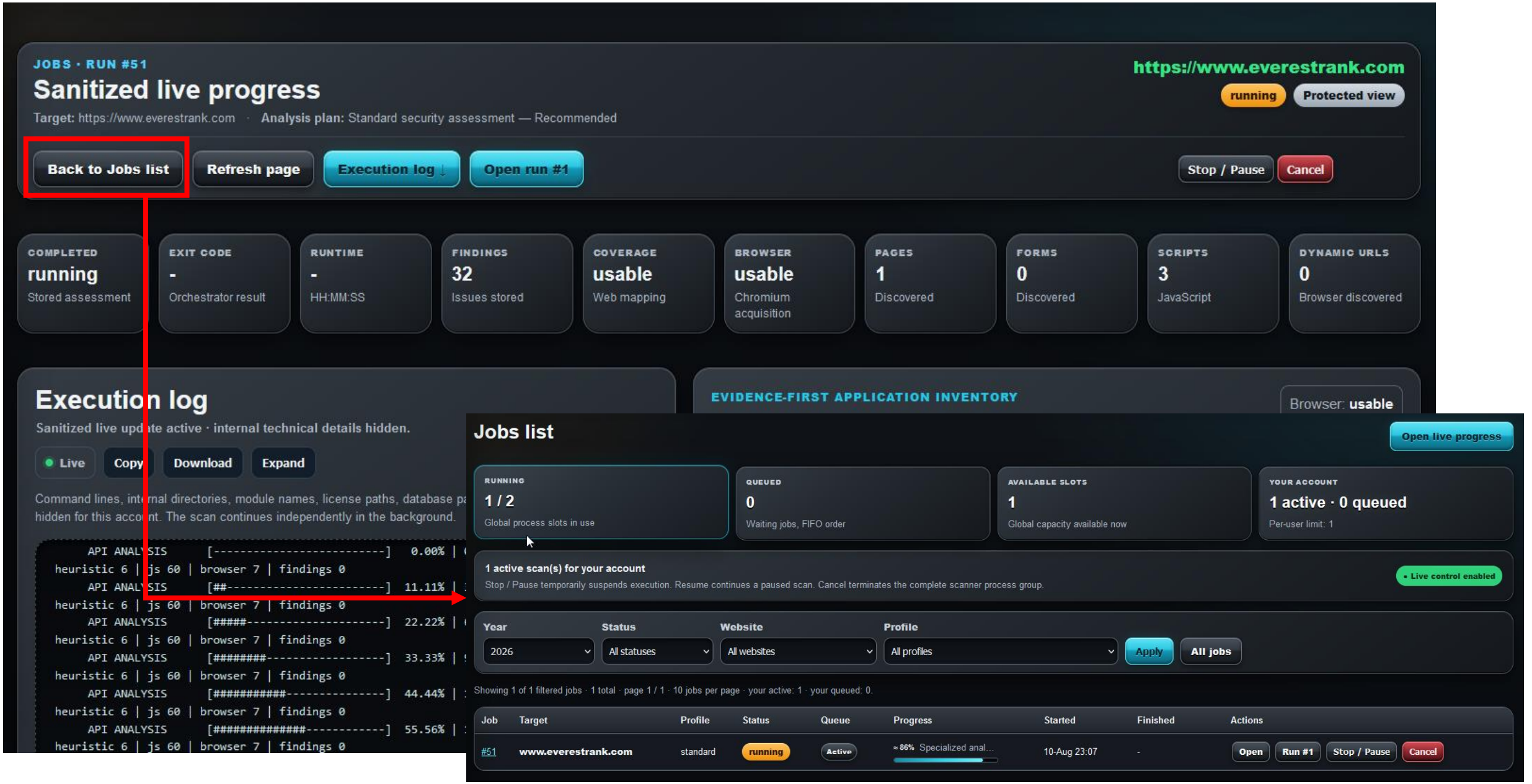


ON PROGRESS LIVE MODAL

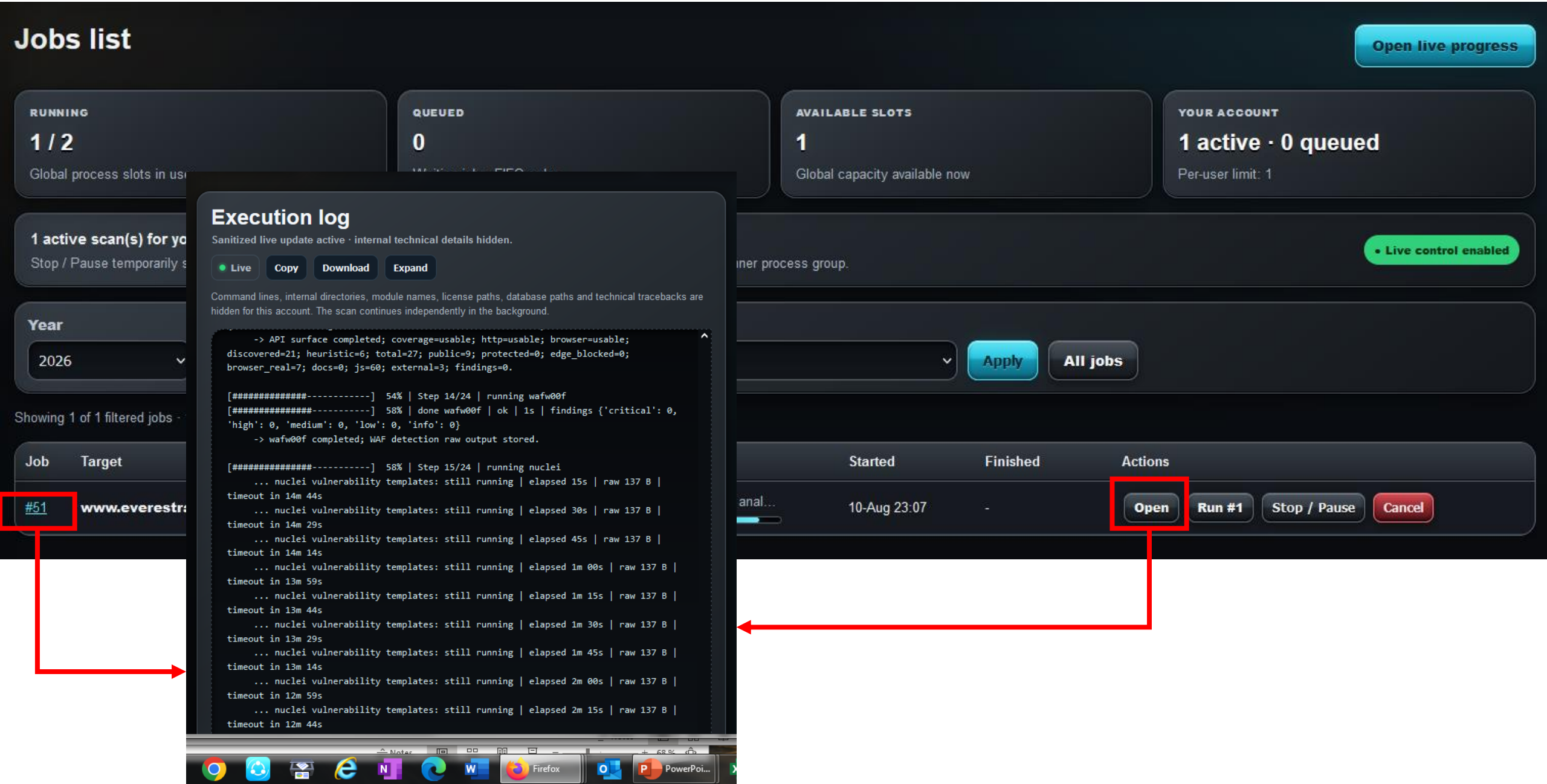




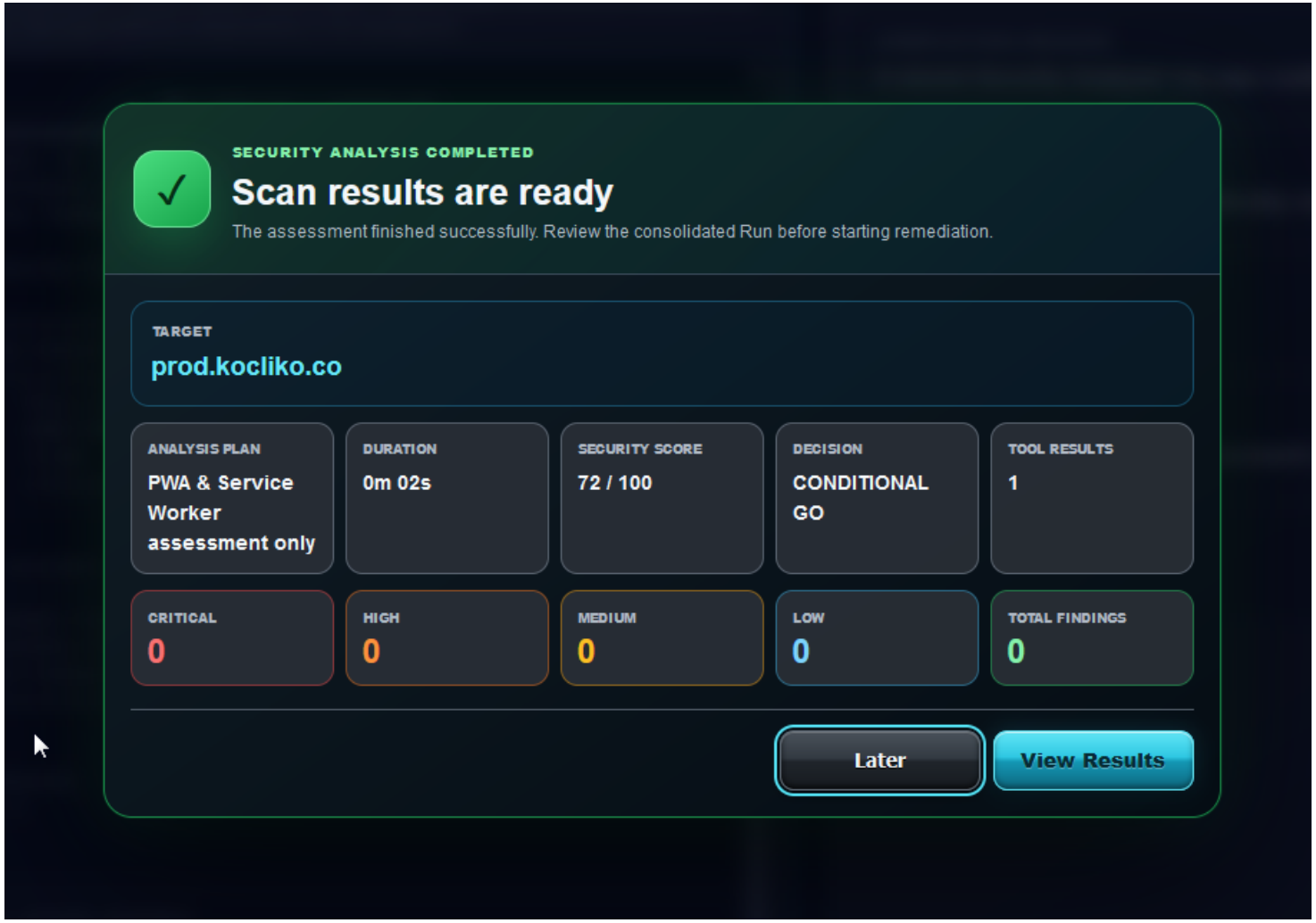
FLIP-FLOP « Live progress » - « Job’s Listing »



FLIP-FLOP « Job's Listing » - « Live progress »



JOB TERMINATED



4. Site Overview

Site overview

5. Scan Job

PROBLEMS

1. Erreurs de formulaire claires

À corriger impérativement.

Chaque champ invalide doit afficher immédiatement :

- une bordure rouge ;
- un message rouge sous le champ ;
- une icône ou un indicateur explicite ;
- un résumé général en haut du formulaire ;
- le focus automatique sur la première erreur.

Exemples :

- `Website URL is required.`
- `Enter a complete website address, for example https://example.com.`
- `Select a scan profile.`
- `You must confirm that you own the target or are explicitly authorized.`

La petite infobulle native du navigateur est insuffisante et différente selon Firefox, Chrome ou Safari.

2. Aucun rechargement en cas d'erreur

Confirmé.

Le formulaire doit être soumis par `fetch` /AJAX et recevoir une réponse JSON.

En cas d'erreur :

- la page ne bouge pas ;
- le formulaire reste visible ;
- les valeurs saisies sont conservées ;
- les messages apparaissent dans les bons champs ;
- aucun job n'est créé ;
- le bouton redevient actif.

Pendant la requête, le bouton doit afficher par exemple :

```
Validating target…
```

et être temporairement désactivé pour éviter les doubles clics.

La validation côté serveur reste obligatoire, même avec la validation JavaScript.

3. Contrôle sérieux de l'URL et de l'existence du site

Confirmé, et c'est également un sujet de sécurité.

La validation doit comporter deux niveaux.

Validation immédiate

Vérifier :

- présence de l'URL ;
- schéma `http://` ou `https://`, ou ajout contrôlé de `https://` ;
- hostname valide ;
- absence d'espaces ou de caractères interdits ;
- port valide ;
- normalisation des slashs et de la casse du hostname.

Précontrôle serveur

Avant de créer le job :

- résolution DNS ;
- rejet des adresses locales, privées, loopback, link-local et réservées ;
- protection contre les attaques SSRF ;
- tentative de connexion avec un délai court ;
- suivi limité et sécurisé des redirections ;
- présentation de l'URL finale ;
- détection des erreurs DNS, connexion refusée, timeout et erreur TLS.

4. Redirection immédiate vers « Live progress »

Confirmé.

Après acceptation du scan, l'API doit retourner quelque chose comme :

JSON

```
{
  "ok": true,
  "job_id": 12,
  "status": "queued",
  "redirect_url": "/job?id=12"
}
```

Le navigateur doit ensuite ouvrir directement la page du job, au niveau du bloc **Live progress**.

Le comportement doit fonctionner que le job soit :

- `queued` ;
- `starting` ;
- `running` .

5. Scan déjà en cours

Confirmé. Le comportement actuel est beaucoup trop silencieux.

Deux situations doivent être distinguées.

Même cible déjà active

Ne créer aucun doublon. Afficher clairement :

```
A scan of https://example.com is already running.
Job #12 — 46% — TLS analysis
```

Avec un bouton :

```
Open Live progress
```

Autre cible déjà active

Comme le système possède une file d'attente et une limite par utilisateur, le nouveau job peut être accepté, mais il faut annoncer :

```
Your account already has one active scan.
This new scan has been placed in the queue at position #2.
```

Puis ouvrir sa page de suivi. Il ne faut jamais donner l'impression que deux scans s'exécutent simultanément lorsqu'un seul slot utilisateur est autorisé.

6. Bouton « Scan Jobs » sur la page Scan Jobs

Confirmé.

La liste doit posséder une ancre dédiée, par exemple :

```
/jobs#scanJobsList
```

Le clic doit systématiquement :

- descendre jusqu'à la liste ;
- placer la liste sous la barre supérieure fixe ;
- produire un bref contour visuel ;
- fonctionner également lorsque l'utilisateur est déjà sur `/jobs`.

L'ancre doit cibler précisément la section contenant les jobs, pas un point générique situé après le bandeau supérieur.

7. Un scan annulé ou incomplet ne doit rien débloquer

Absolument confirmé. C'est actuellement le défaut le plus grave.

La capture montre :

- `60 %` ;
- `9 views ready` ;
- un run comptabilisé ;
- six findings disponibles ;

alors que l'unique tentative a été annulée ou incomplète.

7. Un scan annulé ou incomplet ne doit rien débloquer

Absolument confirmé. C'est actuellement le défaut le plus grave.

La capture montre :

- `60 %` ;
- `9 views ready` ;
- un run comptabilisé ;
- six findings disponibles ;

alors que l'unique tentative a été annulée ou incomplète.

La présence d'une ligne `security_scan_run`, de quelques findings ou d'un `run_id` ne suffit pas à constituer un scan valide.

Un scan ne doit débloquer les vues que lorsqu'il respecte au minimum :

- job `completed` ;
- run finalisé ;
- `finished_at` renseigné ;
- absence de statut `cancelled`, `failed` ou `interrupted` ;
- confirmation de la phase finale de l'orchestrateur ;
- données de synthèse produites ;
- association cohérente entre le job et le run.

Les données partielles peuvent être conservées pour diagnostic, mais elles ne doivent pas alimenter :

8. Place des scans annulés ou incomplets dans l'historique

Voici la règle correcte.

Dans « Scan Jobs »

Oui, toujours.

Une tentative annulée, échouée ou interrompue doit rester visible pour assurer la traçabilité :

- cible ;
- profil ;
- statut ;
- dates ;
- progression atteinte ;
- raison de fin ;
- code de sortie ;
- journal technique.

Dans « Runs » et les sélecteurs métier

Non, pas comme un véritable scan sélectionnable.

Elle ne doit pas être proposée comme baseline normale dans :

- Summary ;
- Compare ;
- Timeline ;
- Portfolio ;
- Compliance ;

9. Plusieurs scans valides

Lorsque plusieurs scans réellement valides existent :

- le premier scan valide débloque Runs, Summary, Findings, Attack, Portfolio, Compliance, Report, Remediation et Risks selon les données réellement produites ;
- Timeline et Compare ne se débloquent qu'avec au moins **deux scans valides du même site canonique** ;
- deux scans de sites différents ne sont pas comparables ;
- si les profils ou la couverture des outils diffèrent fortement, la comparaison doit afficher un avertissement de comparabilité.

Ordre des corrections

Il faut maintenant traiter cela en trois ensembles cohérents :

1. **Formulaire AJAX, messages inline, contrôle URL et redirection Live progress**
2. **Annonce des jobs actifs et navigation/scroll vers la liste**
3. **Classification stricte des runs valides, déverrouillage et séparation de l'historique incomplet**

Ensuite, la recette devra repartir depuis un compte ne possédant aucun scan valide, avec les anciennes tentatives annulées correctement classées comme incomplètes.

PATCH TEST-LAUNCH-UX-002A — Formulaire AJAX et navigation guidée

Patch réalisé à partir de `security_dashboard_app(168).py` . security_dashboard_app

Il corrige les points **1 à 6** :

- erreurs rouges précises sous chaque champ ;
- suppression des infobulles natives du navigateur ;
- aucune recharge de page en cas d'erreur ;
- validation AJAX/XHR ;
- contrôle du format, du DNS, de l'adresse publique et de l'accessibilité du site ;
- blocage des adresses privées, locales et réservées ;
- détection d'un scan identique déjà actif ;
- annonce d'un autre scan déjà en cours et mise en file d'attente ;
- redirection automatique vers **Live progress** ;
- bouton **Scan Jobs** qui descend désormais jusqu'à la liste des jobs ;
- indication des scans déjà actifs directement sur le formulaire.

TO DO

TO DO LIST

- Statut optionnel user (public , privé)
- Si Profile public/visible : Option d’exchange n => +NNN (via messagerie websocket)
- Option « forgot password »
- Option dans account dans la section « account ou admin »
- Fonction d’analyse en « retro engineerie » des Js script actifs
- Analyse / scanner des appels xhr (depuis commandes ajax, js, et autre)

SECURITY GUIDE BY IDEO-LAB

JULY 2026

AUTEUR : GUILLAUME ONEILL

PLAN

- 1. SCRIPT INSTALLATION DE TOUS LES LOGICIELS DE TESTS DE ROBUSTESSE D'UN SITE INTERNET
- 2. PILOTAGE ET ORCHESTRATION DU LANCEMENT DE CHAQUE TEST
- 3. MESSAGES GENERAUX DE PROGRESSION ET BARRE DE PROGRESSION
- 4.EVITER LE VERBOSE CEST ILLISBLE ET PEU INTERESSANT!!!
- 4B. MINI SYNTHESE A CHAQUE OUTIL DE TEST QUI S'EXECUTE
- 5. ANALYSER EN TEMPS L'output de chaque Outils en faire une synthèse
- 6. STOCKER LE RESULTAT ET LE DIAGNOSTIQUE DE CHAQUE OUTIL SOUS SQL
- 7. PREPARER UNE SYNTHESES GLOBALE ET GENERER UN PDF EN SORTIE
- 8. DEVELOPPER LE SCRIPT EN PYTHON
- 9. Auto installer un VENV, car certains addons pour l'installation et execution
- 10. Un système pour arréter proprement le scan en cours d'éxecution
- 11. Un système pour le reprendre (« resume »)
- 12. Une fonction pour lister toutes les execution du scan (date, heure, durée et score)
- 13. Une fonction pour re-afficher telle ou telle analyse réalisée dans le passé

EXTENDED FUNCTIONS

- Capter l'IP adresse
- Capter le nom de l'url scannée
- Demander une adresse email (comme le fait certbot)
- Stocker cela bien sûr sur le SQL report
- Envoyer les données principales (Ip, email, Url, score) sur mon server
  - via une API genre « https://www.ideo-lab.com/security/scan?xxxx= (objet json)
  - Il faudra développer ensuite un micro service sur Ideo-lab
- Prévoir un « aut-update » en option !!!

STARTING PROCEDURES

Procédure simple

Créer le fichier :

Bash

```
nano ideolab_security_orchestrator.py
```

Coller le script ci-dessous, puis :

Bash

```
chmod +x ideolab_security_orchestrator.py
sudo python3 ideolab_security_orchestrator.py install
```

Lancer un audit autorisé :

Bash

```
python3 ideolab_security_orchestrator.py audit --target https://www.ideo-lab.com --i-own
```

Security Orchestrator v1.2

Author : Guillaume Oneill

contact@ideo-lab.com

https://www.ideo-lab.com

Token : xxx xxx xxx

USER GUIDE

Bash

```
python3 ideolab_security_orchestrator.py list-runs
python3 ideolab_security_orchestrator.py show-run --run-id 6
python3 ideolab_security_orchestrator.py report --run-id 6
```

SCAN PROFILE

Bash

```
python3 -m py_compile ideolab_security_orchestrator.py
```

Puis test ciblé rapide :

Bash

```
lit --i-own-this-target --profile quick --only headers,testssl,nuclei --report-mode short
```

Pour tester aussi `ffuf` et `zap` sans relancer tout :

Bash

```
python3 ideolab_security_orchestrator.py audit --i-own-this-target --profile standard --
```

DYNAMIC SETUP

```
python3 ideolab_security_orchestrator.py setup
python3 ideolab_security_orchestrator.py audit --i-own-this-target
python3 ideolab_security_orchestrator.py audit --i-own-this-target --report-mode s
python3 ideolab_security_orchestrator.py config-show
```

ROADMAP DES PATCHS

AGENDA DES PATCHS

- PATCH 012 — Profils de scan quick / standard / deep
- PATCH 013 — Resume automatique / scan interrompu
- PATCH 014 — Install Doctor / Health Check
- PATCH 015 — Meilleurs parsers des outputs outils
- PATCH 016 — Rapport HTML local en plus du PDF
- PATCH 017 — Comparaison entre deux runs
- PATCH 018 — Retest ciblé
- PATCH 019 — Cron / scheduled audit
- PATCH 020 — Packaging propre

Ensuite la suite logique serait :

```
PATCH 019 — Retest ciblé
Relancer automatiquement seulement les outils liés aux findings du run précédent.

PATCH 020 — Cron / scheduled audit
Installer un audit périodique avec rapport email.

PATCH 021 — Packaging propre
Transformer le script en package installable / commande globale.
```

PATCH 012 — Profils de scan `quick / standard / deep`

C'est le plus important maintenant. Aujourd'hui le scan lance quasiment tout. Il faut pouvoir faire :

```bash
python3 ideolab_security_orchestrator.py audit --i-own-this-target --profile quick
python3 ideolab_security_orchestrator.py audit --i-own-this-target --profile standard
python3 ideolab_security_orchestrator.py audit --i-own-this-target --profile deep
```

Avec une logique simple :

```
quick    = nmap rapide + headers + TLS rapide + nuclei léger
standard = scan actuel raisonnable
deep     = nmap full + testssl complet + ffuf large + ZAP + nikto + code/cloud si
```

PATCH 013 — Resume automatique / scan interrompu

Si le scan est coupé, il ne faut pas tout recommencer. Il faut pouvoir faire :

```bash
python3 ideolab_security_orchestrator.py resume
```

Et le script doit reprendre au dernier outil non terminé, sans système de lock pénible.

PATCH 014 — Install Doctor / Health Check

Un diagnostic clair des outils :

```bash
python3 ideolab_security_orchestrator.py doctor
```

Avec sortie du type :

```
nmap      OK
ffuf      OK
nikto     fallback Git OK
semgrep   venv OK
docker    OK but user may need sudo
smtp      missing host
```

ignorer et passer au contenu

PATCH 015 — Meilleurs parsers des outputs outils

C'est là qu'on rend le diagnostic plus intelligent. Aujourd'hui certains findings sont encore trop génériques. Il faut parser plus finement :

```
ZAP         WARN / FAIL par type
testssl     protocoles, ciphers, certificat, HSTS
ffuf        403 vs 200 vs 301 intelligents
nikto       headers, server leaks, dangerous files
nuclei      severity + template + URL + tags
nmap        ports attendus / inattendus / services sensibles
```

PATCH 016 — Rapport HTML local en plus du PDF

Le PDF est bien, mais un HTML est souvent plus lisible sur serveur :

```
report/security_flight_readiness_run_XX.html
```

Avec cartes, couleurs, tableaux, sections collapsibles, chemins bruts, actions prioritaires.

PATCH 017 — Comparaison entre deux runs

Très utile après correction :

```bash
python3 ideolab_security_orchestrator.py diff --run-id 8 --with-run-id 9
```

Objectif :

```
Resolved findings : 7
New findings      : 2
Still open        : 4
Score evolution   : 62 -> 81
```

PATCH 018 — Retest ciblé

Après correction, ne relancer que les outils concernés :

```bash
python3 ideolab_security_orchestrator.py retest --run-id 8
```

Par exemple, si seuls headers et TLS étaient mauvais, on ne relance que `headers` et `testssl`.

PATCH 019 — Cron / scheduled audit

Créer une commande propre :

```bash
python3 ideolab_security_orchestrator.py cron-install
```

Qui installe un audit hebdomadaire ou quotidien avec envoi email si configuré.

PATCH 020 — Packaging propre

Créer une structure professionnelle :

```
security_orchestrator/
  ideolab_security_orchestrator.py
  README.md
  install.sh
  security_orchestrator.conf.example
```

Puis éventuellement :

```bash
security-orchestrator audit
security-orchestrator report --run-id 8
```

ROADMAP DES REPORTS IMPROVEMENT

AGENDA REPORTS

- Niveau 1 : Executive Summary
- Niveau 2 : Health Dashboard
- Niveau 3 : Detailed Findings
- Niveau 4 : Remediation Plan
- Niveau 5 : Short Mode
- Niveau 6 : Report Database

Niveau 1 : Executive Summary

```
============================================================
IDEO-LAB SECURITY ANALYZER
============================================================

Target:
www.ideo-lab.com

Global Score:
82 / 100

Risk Level:
MEDIUM

Critical Issues:
0

High Issues:
2

Medium Issues:
4

Low Issues:
7

============================================================
TOP PRIORITIES
============================================================

[HIGH]
Missing Content-Security-Policy

[HIGH]
Weak TLS configuration

[MEDIUM]
```

Niveau 2 : Health Dashboard

```
WEB SECURITY
-----------------------------------
HTTPS               OK
TLS                 WARNING
HSTS                FAIL
CSP                 FAIL
Cookies             OK

Score : 65/100
```

Puis :

```
SERVER SECURITY
-----------------------------------
Nginx               OK
Version Leak        WARNING
Directory Listing   OK
Compression         OK

Score : 80/100
```

Puis :

```
APPLICATION SECURITY
-----------------------------------
Admin Exposure      OK
Debug Mode          OK
Secrets             NOT TESTED
Authentication      PARTIAL
```

Niveau 3 : Detailed Findings

```
============================================================
FINDING #3
============================================================

Category:
HTTP Security Headers

Severity:
HIGH

Issue:
Missing Content-Security-Policy

Impact:
Potential XSS attack surface.

Recommendation:
Add a strict CSP header in Nginx.

Example:

add_header Content-Security-Policy
"default-src 'self';";

Status:
ACTION REQUIRED
```

Niveau 4 : Remediation Plan

```
============================================================
REMEDIATION ROADMAP
============================================================

Priority 1 (1 hour)
-------------------
Add HSTS

Priority 2 (2 hours)
-------------------
Add CSP

Priority 3 (optional)
---------------------
Hide Nginx version

Expected Score:
82 -> 95
```

Niveau 5 : Short Mode

Bash

```
python ideolab_security_orchestrator.py --short
```

qui produirait seulement :

```
Target:
www.ideo-lab.com

Score:
82/100

Critical:
0

High:
2

Top Issues:
- Missing CSP
- Weak TLS

Next Action:
Add CSP header
```

Niveau 6 : Report Database

Bash

```
python ideolab_security_orchestrator.py --scan
```

on stocke tout dans MariaDB.

Puis :

Bash

```
python ideolab_security_orchestrator.py --summary
```

affiche uniquement la synthèse.

Bash

```
python ideolab_security_orchestrator.py --query high
```

affiche seulement les High.

Bash

```
python ideolab_security_orchestrator.py --query tls
```

affiche seulement TLS.

Bash

```
python ideolab_security_orchestrator.py --diff
```

AGENDA PATCHS

Pour moi, la roadmap idéale est :

PATCH 021A → nouveau moteur de reporting structuré (Executive Summary + Dashboard + Findings + Remediation)

PATCH 021B → mode `--short`

PATCH 021C → stockage des findings en base SQLite/MariaDB

PATCH 021D → moteur de requêtes (`--query` , `--summary` , `--diff`)

C'est exactement ce qui a transformé Migrate Safe V2 d'un simple scanner de logs en un outil réellement exploitable au quotidien. Ici, le Security Analyzer est au même point de maturité : l'analyse est déjà bonne, mais il lui manque une couche de restitution et de pilotage.

ROADMAP DU DASHBOARD IMPROVEMENT 2026

AGENDA PATCHS

```
PATCH 022A
Créer la couche SQL + stockage des runs/findings

PATCH 022B
Réduire la console à un summary court

PATCH 022C
Créer mini web app sur port 8400

PATCH 022D
Dashboard complet + filtres + diff
```

TO DO LIST

- Start / STOP / Resume du serveur
- Interaction avec le security analyzer : ideolab_security_orchestrator
- Pourvoir lancer l’orchestrator/Analyzer à partir du Dashboard
- Historic des runs visible aussi dans le dashboard pour un user connecté

PATCH 025A — Dashboard runtime & logging stabilization

Objectif :

```
1. Ajouter un vrai logging dashboard
2. Ne plus crasher sur ConnectionResetError
3. Afficher chaque requête HTTP utile
4. Ajouter /health
5. Ajouter /server-log
6. Préparer une table SQL security_runtime_event
```

Ça corrige immédiatement :

```
- serveur dashboard trop silencieux
- crash bruité quand Firefox coupe la connexion
- impossibilité de savoir si le dashboard vit correctement
```

Ensuite :

PATCH 025B — Orchestrator control plane

```
- table security_runtime_command
- table security_runtime_state
- dashboard peut demander start / stop / resume
- orchestrator lit les commandes
- statut visible côté dashboard
```

LICENSE KEY PROCESS

LICENSE KEY PROCESS

- python3 security_dashboard_app.py --license-status
- python3 security_dashboard_app.py --license-setup

```
1. Request envoyée au License Center
2. Email de vérification reçu
3. Lien /api/license/confirm/<token>/ cliqué
4. Email verified
5. License key créée
6. Le client a récupéré la clé via polling
7. Activation machine réussie
8. Cache local créé :
   /home/ubuntu/.config/ideolab/security_analyzer_license.json
9. Status = VALID
```

```
ubuntu@ip-172-31-42-227:/home/gon2000ff$ python3 security_dashboard_app.py --license-setup

+-----------------------------------------------------------------------------------------------------------+
| IDEO-Lab Security Analyzer - License Setup                                                                |
|                                                                                                           |
| Product        : ideolab_security_analyzer                                                                |
| Version        : 1.25                                                                                     |
| Machine ID     : f8963a29dec69fe04520f4...                                                                |
| Hostname       : ip-172-31-42-227                                                                         |
| License file   : /home/ubuntu/.config/ideolab/security_analyzer_license.json                              |
|                                                                                                           |
| This assistant will request a license key from Ideo-Lab License Center,                                   |
| then activate the dashboard and save the local activation cache.                                          |
+-----------------------------------------------------------------------------------------------------------+

Full name: oneill
Email: contact@ideo-lab.com
Company: ideolab
License type [trial]:
Reason [Evaluate IDEO-Lab Security Analyzer]:

+-----------------------------------------------------------------------------------------------------------+
| License request summary                                                                                   |
|                                                                                                           |
| Name           : oneill                                                                                   |
| Email          : contact@ideo-lab.com                                                                     |
| Company        : ideolab                                                                                  |
| Product        : ideolab_security_analyzer                                                                |
| License type   : trial                                                                                    |
| Machine hash   : f8963a29dec69fe04520f4...                                                                |
| Request URL    : https://www.ideo-lab.com/api/license/request/                                            |
| Activate URL   : https://www.ideo-lab.com/api/license/activate/                                           |
+-----------------------------------------------------------------------------------------------------------+

Send request now? yes/no [yes]: yes

+-----------------------------------------------------------------------------------------------------------+
| License request created.                                                                                  |
|                                                                                                           |
| Request ID : CLR-BA0598EE9D1349188E                                                                       |
| Email      : contact@ideo-lab.com                                                                         |
|                                                                                                           |
| A verification email has been sent.                                                                       |
| Please click the confirmation link.                                                                       |
| This terminal will wait for the license key.                                                              |
+-----------------------------------------------------------------------------------------------------------+

Waiting for email verification. Please click the link sent to your email.
```

Verify your IDEO-Lab Security Analyzer activation

C

contact@ideo-lab.com
Verify your IDEO-Lab Security Analyzer activation
À contact@ideo-lab.com

11/7/2026 15:37

IDEO-Lab Security Analyzer activation

A license activation request was created for:

Name : oneill
Email : contact@ideo-lab.com
Company : ideolab
Product : ideolab_security_analyzer
Request : CLR-BA0598EE9D1349188E

To confirm this email address and continue activation, open this link:

https://www.ideo-lab.com/api/license/confirm/iyWNJIoB60BhMxNC1pGtvFnQrqn1T_AVFFJ1cDX9ZlI/

This verification link expires in 24 hour(s).

If you did not request this activation, ignore this email.

IDEO-Lab
https://www.ideo-lab.com

Email verified

IDEO-Lab Security Analyzer

Your professional email address has been verified. License request CLR-BA0598EE9D1349188E is now approved. License key status: created.

You may now return to the terminal license setup.

```
+-------------------------------------------------------------------------------+

Waiting for email verification. Please click the link sent to your email.

+-------------------------------------------------------------------------------+
| License key received.                                                         |
|                                                                               |
| Request ID  : CLR-BA0598EE9D1349188E                                          |
| Product     : ideolab_security_analyzer                                       |
| License key : CL2-TRIAL-20260711-4BD021BDDEC6206C-49188E-BE977A8B1FA6         |
| Valid until : 2026-08-10T13:40:08.750613+00:00                                |
|                                                                               |
| Activating this machine now...                                                |
+-------------------------------------------------------------------------------+


+-------------------------------------------------------------------------------+
| License setup completed successfully.                                         |
|                                                                               |
| License activated for ideolab / edition=trial / status=VALID                  |
|                                                                               |
| License cache : /home/ubuntu/.config/ideolab/security_analyzer_license.json   |
|                                                                               |
| You can now start the dashboard with:                                         |
| python3 security_dashboard_app.py --host 0.0.0.0 --port 8400                  |
+-------------------------------------------------------------------------------+
```

ORCHESTRATOR PROCESS

```
ubuntu@ip-172-31-42-227:/home/gon2000fr$ python3 ideolab_security_orchestrator.py list-runs

+-----------------------------------------------+
|          Security Orchestrator   v1.55        |
|             Copyright Ideo-Lab 2026           |
|            Author : Guillaume Oneill          |
|               contact@ideo-lab.com            |
|              https://www.ideo-lab.com         |
|      Token : FF3A 670C 9556 0835 B285 2ECA     |
+-----------------------------------------------+

Press ENTER to continue, or wait 20 seconds.



================================================================================================================================
IDEO-Lab Security Orchestrator - Run history
================================================================================================================================
ID     Decision          Score    Status              Started                 Target
--------------------------------------------------------------------------------------------------------------------------------
48     NO-GO             33       completed           2026-07-11T12:44:24Z    https://www.ideo-lab.com
47     NO-GO             28       completed           2026-07-10T18:32:02Z    https://prod.kocliko.co
46     NO-GO             33       completed           2026-07-07T12:17:23Z    https://www.ubikasec.com
45     NO-GO             33       completed           2026-07-07T11:20:37Z    https://www.ubikasec.com
44     NO-GO             33       completed           2026-07-06T09:40:33Z    https://www.ubikasec.com
43     NO-GO             33       completed           2026-06-30T20:43:46Z    https://www.ideo-lab.com
42     NO-GO             0        completed           2026-06-30T17:36:09Z    https://www.ideo-lab.com
41     NO-GO             0        completed           2026-06-30T15:01:50Z    https://www.ideo-lab.com
40     NO-GO             0        completed           2026-06-19T09:44:58Z    https://www.ideo-lab.com
39     NO-GO             0        completed           2026-06-19T09:40:03Z    https://translate.ideo-lab.com
38     -                 -        running             2026-06-19T09:37:34Z    https://translate.ideo-lab.com
37     NO-GO             0        completed           2026-06-18T18:59:24Z    https://www.ideo-lab.com
36     NO-GO             0        completed           2026-06-18T18:54:30Z    https://translate.ideo-lab.com
35     NO-GO             0        completed           2026-06-18T18:24:13Z    https://www.ideo-lab.com
34     NO-GO             0        completed           2026-06-18T18:19:21Z    https://translate.ideo-lab.com
33     CONDITIONAL GO    79       completed           2026-06-18T17:46:02Z    https://www.ideo-lab.com
32     CONDITIONAL GO    79       completed           2026-06-18T17:46:01Z    https://translate.ideo-lab.com
31     CONDITIONAL GO    79       completed           2026-06-18T17:20:47Z    https://ip-172-31-42-227.eu-central-1.compute.internal
30     CONDITIONAL GO    79       completed           2026-06-18T17:20:46Z    https://www.ideo-lab.com
29     CONDITIONAL GO    79       completed           2026-06-18T17:20:46Z    https://translate.ideo-lab.com
================================================================================================================================
```

ANALYZER SETTINGS

DJANGO SETTINGS

```python
# ============================================================
# IDEO-LAB SECURITY ANALYZER — INFRA PUBLIC HTTPS 2026
# ============================================================

IDEO_SECURITY_ANALYZER_PUBLIC_URL = "https://analyzer.ideo-lab.com"
IDEO_SECURITY_LICENSE_PUBLIC_BASE_URL = "https://license.ideo-lab.com"

# Produits autorisés par le License Center.
CRASHLENS_LICENSE_ALLOWED_PRODUCTS = [
    "crashlens-django",
    "ideolab_security_analyzer",
]

# Hosts publics autorisés.
try:
    ALLOWED_HOSTS = list(ALLOWED_HOSTS)
except Exception:
    ALLOWED_HOSTS = []

for _host in [
    "www.ideo-lab.com",
    "ideo-lab.com",
    "staging.ideo-lab.com",
    "analyzer.ideo-lab.com",
    "license.ideo-lab.com",
    "3.72.171.177",
    "127.0.0.1",
    "localhost",
]:
    if _host not in ALLOWED_HOSTS:
        ALLOWED_HOSTS.append(_host)

# Origines HTTPS de confiance.
try:
    CSRF_TRUSTED_ORIGINS = list(CSRF_TRUSTED_ORIGINS)
except Exception:
    CSRF_TRUSTED_ORIGINS = []

for _origin in [
    "https://www.ideo-lab.com",
    "https://ideo-lab.com",
    "https://staging.ideo-lab.com",
    "https://analyzer.ideo-lab.com",
    "https://license.ideo-lab.com",
]:
    if _origin not in CSRF_TRUSTED_ORIGINS:
        CSRF_TRUSTED_ORIGINS.append(_origin)

# Django derrière Nginx HTTPS.
SECURE_PROXY_SSL_HEADER = ("HTTP_X_FORWARDED_PROTO", "https")
USE_X_FORWARDED_HOST = True

# Cookies sécurisés uniquement en HTTPS public.
CSRF_COOKIE_SECURE = True
SESSION_COOKIE_SECURE = True
CSRF_COOKIE_SAMESITE = "Lax"
SESSION_COOKIE_SAMESITE = "Lax"
```

NGINX SETTINGS

```
    listen [::]:80;
    server_name analyzer.ideo-lab.com;

    location ^~ /.well-known/acme-challenge/ {
        root /var/www/letsencrypt;
        default_type "text/plain";
        try_files $uri =404;
    }

    location / {
        proxy_pass http://127.0.0.1:8400;

        proxy_http_version 1.1;
        proxy_set_header Connection "";

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Proto $scheme;

        proxy_connect_timeout 5s;
        proxy_read_timeout 60s;
        proxy_send_timeout 60s;
    }
}
```

Fonctions connexes très utiles

1. Disponibilité / continuité de service

```
- site online/offline
- code HTTP retourné
- temps de réponse
- certificat encore valide
- redirection HTTP -> HTTPS OK
- contenu minimum présent
- variation du titre HTML
- variation hash de la homepage
- erreurs 5xx
- erreurs DNS
- timeout
```

Périodes :

```
- 24 heures
- 7 jours
- 30 jours
- 90 jours
```

Résultat dashboard :

```
Uptime 24h     : 100%
Uptime 7j      : 99.82%
Uptime 30j     : 99.64%
Incidents      : 3
Temps moyen    : 284 ms
Temps max      : 2.8 s
Dernière panne : 2026-07-10 03:14
```

Ça nécessite un mini scheduler côté dashboard ou un service dédié.

2. DNS / domaine / infrastructure publique

Analyse totalement distante :

```
- A / AAAA
- CNAME
- MX
- TXT
- SPF
- DKIM probable
- DMARC
- CAA
- NS
- TTL
- DNSSEC
- WHOIS / RDAP si disponible
- date d'expiration domaine
- registrar
- nameservers
- IP publiques
- ASN / hébergeur
- géolocalisation approximative IP
```

Très utile pour un rapport client.

Exemples de diagnostics :

```
DMARC absent
SPF trop permissif
CAA absent
TTL trop bas ou incohérent
Domaine proche expiration
MX exposé sans politique DMARC
```

3. SEO technique

```
- title présent / longueur correcte
- meta description présente
- canonical
- robots.txt
- sitemap.xml
- hreflang
- balises H1/H2
- OpenGraph
- Twitter Card
- schema.org JSON-LD
- viewport mobile
- alt images
- liens cassés internes
- profondeur des liens
- pages indexables / noindex
- redirections multiples
- 404 importantes
```

Score possible :

```
SEO technical score : 72/100
```

Avec recommandations :

```
- Ajouter meta description
- Corriger canonical
- Ajouter sitemap.xml
- Réduire les redirections
- Ajouter OpenGraph image
```

4. Visibilité moteurs de recherche

```
Mode sans API :
- robots.txt
- sitemap.xml
- balises noindex/nofollow
- canonical
- présence sitemap
- pages déclarées dans sitemap
- pages accessibles
- structure SEO

Mode avec API optionnelle :
- Bing Web Search API
- Google Custom Search API
- SerpAPI / DataForSEO si le client fournit une clé
```

Données utiles :

```
- nombre approximatif de pages indexées
- requêtes de marque
- présence homepage sur requête marque
- titres visibles dans SERP
- snippets
- pages mal indexées
```

Dans le dashboard :

```
Search visibility
    Brand query visible : YES
    Homepage found      : YES
    Indexed sample      : 42 pages
    Sitemap URLs        : 128
    Noindex pages       : 3
```

5. Réputation sécurité / blacklist

Analyse distante très utile :

```
- Google Safe Browsing API
- VirusTotal URL/domain API
- URLhaus
- PhishTank
- Spamhaus / DNSBL
- OpenPhish
- abuse.ch
```

Score :

```
Reputation status : CLEAN / WARNING / LISTED
```

Findings :

```
- Domaine listé dans une blacklist
- IP signalée comme abusive
- URL détectée dans une base phishing
```

Ça donne énormément de valeur au produit.

6. Presse / actualités / marque

Sources possibles :

```
- RSS public du site
- flux news configurés
- Bing News API
- GDELT
- Google News via source externe/API
- recherche web API
```

Objectif :

```
- mentions récentes de la marque
- sentiment approximatif
- sujets associés
- signaux de crise
- articles récents
- fréquence de publication
```

Dashboard :

```
Brand / News monitoring
    Mentions last 7d  : 12
    Positive          : 5
    Neutral           : 6
    Negative          : 1
    Last mention      : ...
```

7. Performance web

Sans être un clone de Lighthouse, on peut déjà faire :

```
- temps DNS
- temps connect
- temps TLS
- time to first byte
- taille HTML
- taille homepage
- nombre assets CSS/JS/images
- compression gzip/brotli
- cache-control
- CDN apparent
- images lourdes
```

Puis en niveau 2 :

```
- Lighthouse CLI
- Core Web Vitals via PageSpeed API
- mobile score
- accessibility score
- best practices score
```

8. Accessibilité / conformité basique

Analyse distante HTML :

```
- images sans alt
- boutons/liens sans texte
- absence lang HTML
- contrastes difficiles à vérifier sans navigateur, mais possible plus tard
- titres hiérarchiques incohérents
- formulaires sans label
```

Très utile pour agences web / clients corporate.

9. Analyse contenu / qualité éditoriale

9. Analyse contenu / qualité éditoriale

À distance :

```
- langue détectée
- longueur texte homepage
- densité contenu réel vs HTML
- duplicate title
- duplicate meta description
- pages vides
- pages pauvres
- textes trop courts
- présence mentions légales
- privacy policy
- terms
- contact page
```

10. Détection changement / défiguration

Très utile sécurité + supervision :

```
- hash homepage
- titre HTML
- taille HTML
- mots clés attendus
- capture textuelle
- changement brutal de contenu
- apparition mots suspects : hacked, casino, viagra, bitcoin, etc.
```

Cela donne un module :

```
Defacement Watch
```

Organisation produit proposée

Dans le dashboard, on pourrait avoir à terme :

```
New Scan
Jobs
Dashboard
Runs
Findings
Remediation
Diff
Health
Log
Cleanup

+ Site Intelligence
+ Uptime
+ SEO
+ Search Visibility
+ DNS / Domain
+ Reputation
+ News / Mentions
```

Mais côté code, il ne faut pas tout mélanger dans `audit`.

Je propose de créer un **nouveau mode orchestrateur** :

Bash

```
python3 ideolab_security_orchestrator.py intel --target https://www.example.com --i-o
```

Puis plus tard :

Bash

```
python3 ideolab_security_orchestrator.py monitor --target https://www.example.com --i
```

Roadmap propre - Patchs

PATCH INTEL-001 — Site Intelligence baseline

```
- DNS A/AAAA/CNAME/MX/TXT/NS
- SPF / DMARC
- robots.txt
- sitemap.xml
- title
- meta description
- canonical
- h1/h2
- OpenGraph
- Twitter Card
- schema.org JSON-LD
- viewport mobile
- basic performance curl timing
- HTTP status map homepage / robots / sitemap
```

Stockage SQLite :

```
site_intel_run
site_intel_metric
site_intel_finding
```

Dashboard :

```
/intel
/intel-run?id=N
```

PATCH INTEL-002 — Uptime monitor

```
- table uptime_probe
- probe toutes les 5 minutes
- status online/offline
- temps réponse
- incident start/end
- uptime 24h / 7j / 30j
```

Dashboard :

```
/uptime
/uptime-target?id=N
```

PATCH INTEL-003 — Reputation / blacklist

```
- Safe Browsing si API key
- VirusTotal si API key
- URLhaus public
- DNSBL basique IP/domain
- statut CLEAN/WARNING/LISTED
```

PATCH INTEL-004 — Search visibility

```
- mode sans API : indexabilité technique
- mode avec API : Bing / Google Custom Search / SerpAPI
- requête marque
- pages trouvées
- snippets
```

PATCH INTEL-005 — News / press / brand monitoring

```
- RSS
- GDELT
- News API optionnelle
- mentions marque
- sentiment simple
- timeline
```

systemd timer multi-target

Créer le fichier de configuration

```bash
sudo mkdir -p /etc/ideolab
sudo nano /etc/ideolab/security-uptime.env
```

[Unit]
Description=IDEO-Lab Security Analyzer uptime multi-target probe
After=network-online.target
Wants=networkonline.target
[Service]
Type=oneshot
User=ubuntu
WorkingDirectory=/home/gon2000fr
EnvironmentFile=-/etc/ideolab/security-uptime.env
ExecStart=/bin/bash -lc '/usr/bin/python3 /home/gon2000fr/ideolab_security_orchestrator.py uptime --targets "$IDEO_SECURITY_UPTIME_TARGETS" --i-own-this-target --db "$IDEO_SECURITY_UPTIME_DB" --license-file "$IDEO_SECURITY_LICENSE_FILE"'

Créer le service systemd

Bash

```
sudo nano /etc/systemd/system/security-uptime-probe.service
```

```
[Unit]
Description=IDEO-Lab Security Analyzer uptime multi-target probe
After=network-online.target
Wants=network-online.target

[Service]
Type=oneshot
User=ubuntu
WorkingDirectory=/home/gon2000fr
EnvironmentFile=-/etc/ideolab/security-uptime.env
ExecStart=/bin/bash -lc '/usr/bin/python3 /home/gon2000fr/ideolab_security_orchestrator.
```

Créer le timer

Bash

```
sudo nano /etc/systemd/system/security-uptime-probe.timer
```

INI

```
[Unit]
Description=Run IDEO-Lab uptime multi-target probe every 5 minutes

[Timer]
OnBootSec=2min
OnUnitActiveSec=5min
Unit=security-uptime-probe.service

[Install]
WantedBy=timers.target
```

Activer le timer

3. Activer le timer

Bash

```bash
sudo systemctl daemon-reload
sudo systemctl enable --now security-uptime-probe.timer
systemctl list-timers | grep security-uptime
```

---

4. Lancer une exécution immédiate

Bash

```bash
sudo systemctl start security-uptime-probe.service
sudo systemctl status security-uptime-probe.service --no-pager -l
```

SECURITY GUIDE BY IDEO-LAB

JULY 2026

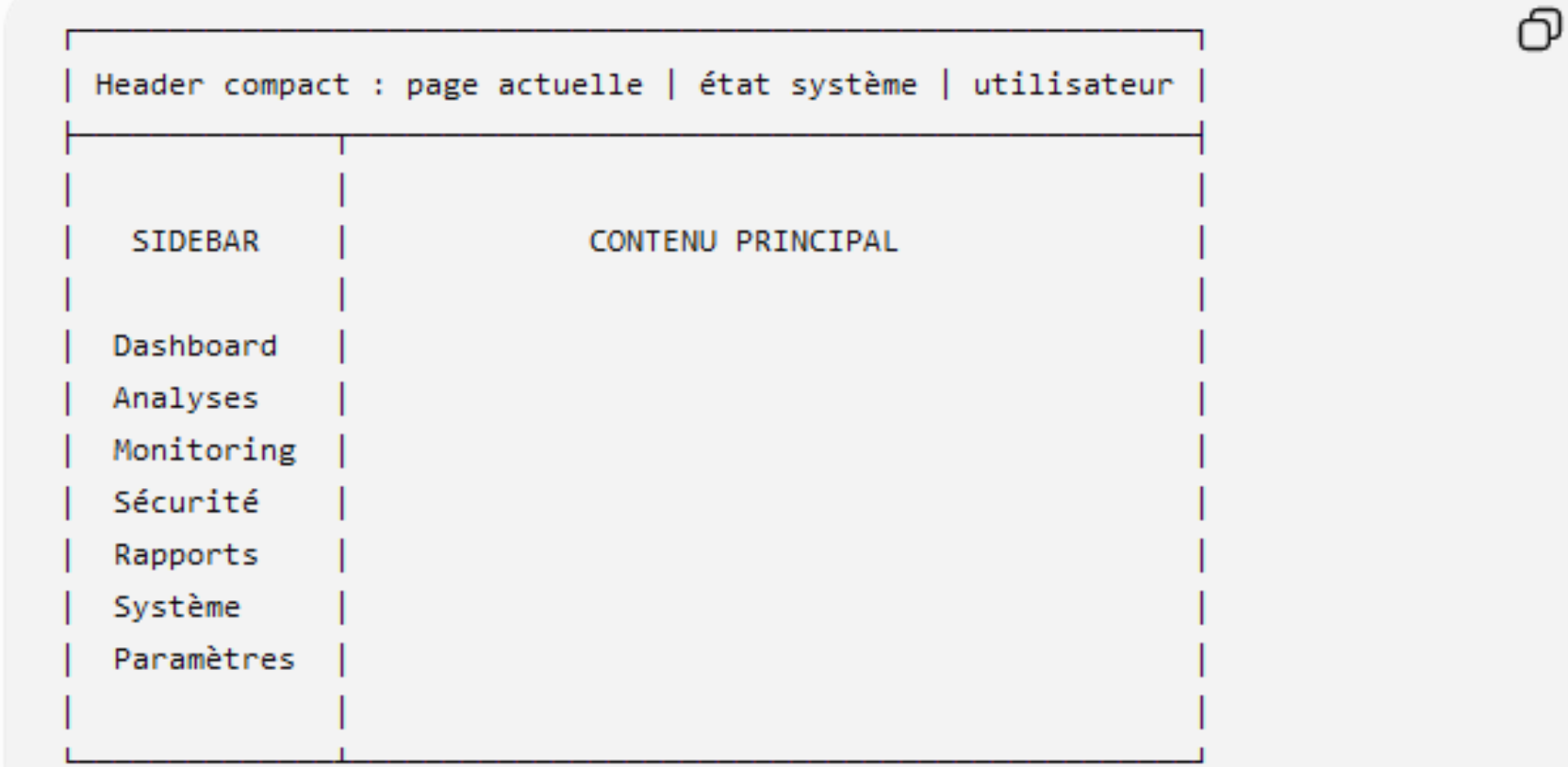
AUTEUR : GUILLAUME ONEILL

MENU LATERAL

PATCH DASHBOARD-UX-002 — Sidebar SaaS rétractable

La navigation principale doit être déplacée dans une **barre latérale verticale gauche**, permanente sur ordinateur et escamotable sur les petits écrans.

Structure générale retenue



Comportement attendu

Mode étendu

Mode contracté

Seules les icônes restent visibles :

```
[SA]

▣
⌕
◉
◆
▤
⚙
```

Au survol d'une icône, une infobulle indiquera clairement :

```
Analyses
Monitoring
Sécurité
Rapports
Administration
```

Règles ergonomiques

La sidebar devra :

- pouvoir être ouverte ou fermée manuellement avec un bouton ;
- mémoriser le choix de l'utilisateur avec `localStorage` ;
- réduire automatiquement sa largeur lorsque l'écran devient plus petit ;
- devenir un panneau coulissant sur mobile ;
- mettre clairement en évidence la section active ;
- n'ouvrir qu'un seul groupe de sous-menus à la fois ;
- afficher des badges lorsque cela apporte une information utile ;
- conserver les aides contextuelles introduites dans les patchs précédents.

Exemples de badges :

```
Analyses en cours      2
Incidents              1
Rapports disponibles   6
Alertes système        3
```

Ce qui restera horizontal

Il ne faut pas supprimer tous les éléments horizontaux. Ils doivent simplement être réservés à la navigation locale d'une page.

Par exemple, dans la page **Monitoring** :

```
Vue générale | Sites | Incidents | Historique | Configuration
```

Ou dans une fiche d'analyse :

```
Résumé | Vulnérabilités | TLS | Headers | Réseau | Recommandations
```

Ce qui restera horizontal

Il ne faut pas supprimer tous les éléments horizontaux. Ils doivent simplement être réservés à la navigation locale d'une page.

Par exemple, dans la page **Monitoring** :

```
Vue générale | Sites | Incidents | Historique | Configuration
```

Ou dans une fiche d'analyse :

```
Résumé | Vulnérabilités | TLS | Headers | Réseau | Recommandations
```

La règle devient donc simple :

- **sidebar gauche** : navigation entre les grandes fonctionnalités ;
- **onglets horizontaux** : navigation interne à la fonctionnalité sélectionnée ;
- **boutons** : actions immédiates ;
- **menus déroulants** : options secondaires.

Header supérieur simplifié

Le haut de l'écran ne doit plus contenir toute la navigation. Il pourra uniquement contenir :

```
[☰]  Analyses / Résultat du scan

Serveur : opérationnel
Orchestrateur : inactif
Dernière analyse : il y a 12 min

[Thème] [Aide] [Notifications] [Utilisateur]
```

Le bouton ☰ permettra de contracter ou d'ouvrir la sidebar.

Organisation fonctionnelle proposée

1. Vue d'ensemble

- tableau de bord général ;
- score global ;
- analyses récentes ;
- incidents ;
- état de l'orchestrateur ;
- état des services ;
- disponibilité des sites.

2. Analyses

- lancer une analyse ;
- analyses en cours ;
- résultats ;
- historique ;
- comparaison entre deux runs ;
- analyses planifiées.

3. Monitoring

- supervision uptime ;
- sites surveillés ;
- incidents ;
- temps de réponse ;
- historique de disponibilité ;
- configuration des fréquences.

4. Sécurité

- vulnérabilités ;
- TLS et certificats ;
- headers HTTP ;
- ports ;
- DNS ;
- technologies détectées ;
- recommandations.

5. Rapports

- génération de rapports ;
- rapports HTML ;
- exports PDF ou JSON ;
- historique ;
- partage ;
- planification.

6. Système

- état de l'orchestrateur ;
- services Linux ;
- daemons ;
- tâches planifiées ;
- files d'attente ;
- journaux ;
- diagnostics serveur.

Organisation fonctionnelle proposée 7

7. Administration

- utilisateurs ;
- rôles ;
- abonnements ou quotas ;
- paramètres globaux ;
- configuration SMTP ;
- sécurité applicative.

Aspect visuel

Pour rester cohérent avec ce que tu demandes depuis plusieurs patchs :

- bleu principal conservé ;
- fond de sidebar légèrement plus soutenu que le contenu ;
- bordures visibles ;
- relief léger ;
- icônes lisibles ;
- états actifs très distincts ;
- contraste renforcé ;
- thème clair et thème sombre compatibles ;
- animations courtes et discrètes ;
- pas d'effets gadgets.

Exemple de section active :

```
| ◆  Sécurité          |
```

Découpage du patch

PATCH DASHBOARD-UX-002A — Structure de la sidebar

- conteneur général ;
- sidebar gauche ;
- zone de contenu ;
- header compact ;
- bouton de contraction ;
- mémorisation de l'état.

PATCH DASHBOARD-UX-002B — Navigation et groupes repliables

- catégories ;
- sous-menus ;
- section active ;
- ouverture exclusive d'un groupe ;
- infobulles du mode contracté.

PATCH DASHBOARD-UX-002C — Responsive mobile

- sidebar coulissante ;
- overlay ;
- fermeture tactile ;
- adaptation tablette et mobile.

PATCH DASHBOARD-UX-002D — Nettoyage de l'ancienne navigation

- suppression des doublons horizontaux ;
- conservation uniquement des onglets contextuels ;
- déplacement des boutons au bon niveau ;
- harmonisation des pages existantes.

Dernier Patch

PATCH DASHBOARD-UX-002E — Finitions SaaS

- badges ;
- état des services ;
- notifications ;
- thème clair et sombre ;
- animations ;
- accessibilité clavier ;
- aides contextuelles.

La prochaine intervention logique est donc :

> **PATCH DASHBOARD-UX-002A — Shell SaaS avec sidebar gauche rétractable**

SECURITY GUIDE BY IDEO-LAB

17 JULY 2026

AUTEUR : GUILLAUME ONEILL

v 1,65

PLAN

- 1. SCRIPT INSTALLATION DE TOUS LES LOGICIELS DE TESTS DE ROBUSTESSE D'UN SITE INTERNET
- 2. PILOTAGE ET ORCHESTRATION DU LANCEMENT DE CHAQUE TEST
- 3. MESSAGES GENERAUX DE PROGRESSION ET BARRE DE PROGRESSION
- 4.EVITER LE VERBOSE CEST ILLISBLE ET PEU INTERESSANT!!!
- 4B. MINI SYNTHESE A CHAQUE OUTIL DE TEST QUI S'EXECUTE
- 5. ANALYSER EN TEMPS L'output de chaque Outils en faire une synthèse
- 6. STOCKER LE RESULTAT ET LE DIAGNOSTIQUE DE CHAQUE OUTIL SOUS SQL
- 7. PREPARER UNE SYNTHESES GLOBALE ET GENERER UN PDF EN SORTIE
- 8. DEVELOPPER LE SCRIPT EN PYTHON
- 9. Auto installer un VENV, car certains addons pour l'installation et execution
- 10. Un système pour arréter proprement le scan en cours d'éxecution
- 11. Un système pour le reprendre (« resume »)
- 12. Une fonction pour lister toutes les execution du scan (date, heure, durée et score)
- 13. Une fonction pour re-afficher telle ou telle analyse réalisée dans le passé

EXTENDED FUNCTIONS

- Capter l'IP adresse
- Capter le nom de l'url scannée
- Demander une adresse email (comme le fait certbot)
- Stocker cela bien sûr sur le SQL report
- Envoyer les données principales (Ip, email, Url, score) sur mon server
  - via une API genre « https://www.ideo-lab.com/security/scan?xxxx= (objet json)
  - Il faudra développer ensuite un micro service sur Ideo-lab
- Prévoir un « aut-update » en option !!!

Fonctions connexes très utiles

1. Disponibilité / continuité de service

```
- site online/offline
- code HTTP retourné
- temps de réponse
- certificat encore valide
- redirection HTTP -> HTTPS OK
- contenu minimum présent
- variation du titre HTML
- variation hash de la homepage
- erreurs 5xx
- erreurs DNS
- timeout
```

Périodes :

```
- 24 heures
- 7 jours
- 30 jours
- 90 jours
```

Résultat dashboard :

```
Uptime 24h     : 100%
Uptime 7j      : 99.82%
Uptime 30j     : 99.64%
Incidents      : 3
Temps moyen    : 284 ms
Temps max      : 2.8 s
Dernière panne : 2026-07-10 03:14
```

Ça nécessite un mini scheduler côté dashboard ou un service dédié.

2. DNS / domaine / infrastructure publique

Analyse totalement distante :

```
- A / AAAA
- CNAME
- MX
- TXT
- SPF
- DKIM probable
- DMARC
- CAA
- NS
- TTL
- DNSSEC
- WHOIS / RDAP si disponible
- date d'expiration domaine
- registrar
- nameservers
- IP publiques
- ASN / hébergeur
- géolocalisation approximative IP
```

Très utile pour un rapport client.

Exemples de diagnostics :

```
DMARC absent
SPF trop permissif
CAA absent
TTL trop bas ou incohérent
Domaine proche expiration
MX exposé sans politique DMARC
```

3. SEO technique

```
- title présent / longueur correcte
- meta description présente
- canonical
- robots.txt
- sitemap.xml
- hreflang
- balises H1/H2
- OpenGraph
- Twitter Card
- schema.org JSON-LD
- viewport mobile
- alt images
- liens cassés internes
- profondeur des liens
- pages indexables / noindex
- redirections multiples
- 404 importantes
```

Score possible :

```
SEO technical score : 72/100
```

Avec recommandations :

```
- Ajouter meta description
- Corriger canonical
- Ajouter sitemap.xml
- Réduire les redirections
- Ajouter OpenGraph image
```

4. Visibilité moteurs de recherche

```
Mode sans API :
- robots.txt
- sitemap.xml
- balises noindex/nofollow
- canonical
- présence sitemap
- pages déclarées dans sitemap
- pages accessibles
- structure SEO

Mode avec API optionnelle :
- Bing Web Search API
- Google Custom Search API
- SerpAPI / DataForSEO si le client fournit une clé
```

Données utiles :

```
- nombre approximatif de pages indexées
- requêtes de marque
- présence homepage sur requête marque
- titres visibles dans SERP
- snippets
- pages mal indexées
```

Dans le dashboard :

```
Search visibility
    Brand query visible : YES
    Homepage found      : YES
    Indexed sample      : 42 pages
    Sitemap URLs        : 128
    Noindex pages       : 3
```

5. Réputation sécurité / blacklist

Analyse distante très utile :

```
- Google Safe Browsing API
- VirusTotal URL/domain API
- URLhaus
- PhishTank
- Spamhaus / DNSBL
- OpenPhish
- abuse.ch
```

Score :

```
Reputation status : CLEAN / WARNING / LISTED
```

Findings :

```
- Domaine listé dans une blacklist
- IP signalée comme abusive
- URL détectée dans une base phishing
```

Ça donne énormément de valeur au produit.

6. Presse / actualités / marque

Sources possibles :

```
- RSS public du site
- flux news configurés
- Bing News API
- GDELT
- Google News via source externe/API
- recherche web API
```

Objectif :

```
- mentions récentes de la marque
- sentiment approximatif
- sujets associés
- signaux de crise
- articles récents
- fréquence de publication
```

Dashboard :

```
Brand / News monitoring
    Mentions last 7d  : 12
    Positive          : 5
    Neutral           : 6
    Negative          : 1
    Last mention      : ...
```

7. Performance web

Sans être un clone de Lighthouse, on peut déjà faire :

```
- temps DNS
- temps connect
- temps TLS
- time to first byte
- taille HTML
- taille homepage
- nombre assets CSS/JS/images
- compression gzip/brotli
- cache-control
- CDN apparent
- images lourdes
```

Puis en niveau 2 :

```
- Lighthouse CLI
- Core Web Vitals via PageSpeed API
- mobile score
- accessibility score
- best practices score
```

8. Accessibilité / conformité basique

Analyse distante HTML :

```
- images sans alt
- boutons/liens sans texte
- absence lang HTML
- contrastes difficiles à vérifier sans navigateur, mais possible plus tard
- titres hiérarchiques incohérents
- formulaires sans label
```

Très utile pour agences web / clients corporate.

9. Analyse contenu / qualité éditoriale

9. Analyse contenu / qualité éditoriale

À distance :

```
- langue détectée
- longueur texte homepage
- densité contenu réel vs HTML
- duplicate title
- duplicate meta description
- pages vides
- pages pauvres
- textes trop courts
- présence mentions légales
- privacy policy
- terms
- contact page
```

10. Détection changement / défiguration

Très utile sécurité + supervision :

```
- hash homepage
- titre HTML
- taille HTML
- mots clés attendus
- capture textuelle
- changement brutal de contenu
- apparition mots suspects : hacked, casino, viagra, bitcoin, etc.
```

Cela donne un module :

```
Defacement Watch
```

Organisation produit proposée

Dans le dashboard, on pourrait avoir à terme :

```
New Scan
Jobs
Dashboard
Runs
Findings
Remediation
Diff
Health
Log
Cleanup

+ Site Intelligence
+ Uptime
+ SEO
+ Search Visibility
+ DNS / Domain
+ Reputation
+ News / Mentions
```

Mais côté code, il ne faut pas tout mélanger dans `audit`.

Je propose de créer un **nouveau mode orchestrateur** :

Bash

```
python3 ideolab_security_orchestrator.py intel --target https://www.example.com --i-o
```

Puis plus tard :

Bash

```
python3 ideolab_security_orchestrator.py monitor --target https://www.example.com --i
```

Roadmap propre - Patchs

PATCH INTEL-001 — Site Intelligence baseline

```
- DNS A/AAAA/CNAME/MX/TXT/NS
- SPF / DMARC
- robots.txt
- sitemap.xml
- title
- meta description
- canonical
- h1/h2
- OpenGraph
- Twitter Card
- schema.org JSON-LD
- viewport mobile
- basic performance curl timing
- HTTP status map homepage / robots / sitemap
```

Stockage SQLite :

```
site_intel_run
site_intel_metric
site_intel_finding
```

Dashboard :

```
/intel
/intel-run?id=N
```

PATCH INTEL-002 — Uptime monitor

```
- table uptime_probe
- probe toutes les 5 minutes
- status online/offline
- temps réponse
- incident start/end
- uptime 24h / 7j / 30j
```

Dashboard :

```
/uptime
/uptime-target?id=N
```

PATCH INTEL-003 — Reputation / blacklist

```
- Safe Browsing si API key
- VirusTotal si API key
- URLhaus public
- DNSBL basique IP/domain
- statut CLEAN/WARNING/LISTED
```

PATCH INTEL-004 — Search visibility

```
- mode sans API : indexabilité technique
- mode avec API : Bing / Google Custom Search / SerpAPI
- requête marque
- pages trouvées
- snippets
```

PATCH INTEL-005 — News / press / brand monitoring

```
- RSS
- GDELT
- News API optionnelle
- mentions marque
- sentiment simple
- timeline
```

TO DO LIST - july 2026

A FAIRE ET A PREVOIR

- sur la page admin / uptime :
  - bouton start/stop/resume en face du site monitoré
  - Une icone pour afficher l'état du site internet (avec des rond rouge et vert), facon calendrier
  - Améliorer l'interface de crétation de la tache de Uptime (frequence, durée)

- Help FAQ Online
  - Help détaillé pour chaque RUbrique
  - Des exemples clairs d'utilisations
- Admin tool uniquement pour les membres admin donc moi :
  - créer un nouvel utilisateur super admin : gon2000fr@gmail.com
  - L'admin tool donnera la liste des inscrits / User connectés
  - L'admin tool permettra de se connecter au user (comme super user)

A FAIRE ET A PREVOIR - 2

- Il faut ajouter une clause About : Header page
- Il faut ajouter une clause contact : Header page
- Il faut ajouter une clause Privacy : Footer Page

- Il faut ajouter un footer :
  - contact email : contact@ideo-lab.com
  - Présentation société (Lockheed LLC & Ideo-lab)
  - Adresse

- Une procédure d'auto-test qui permettra de valider que l'ensemble du produit « Security analyzer » fonctionne parfaitement

A FAIRE ET A PREVOIR - 3

- Mettre dans le menu du header un menu (genre notifications, rappel) : (j'aime bien l'icone « cloche » avec en rouge en haut à droite de l'icone, un petit chiffre rouge (du nombre de notifications récentes non lues ou reardées)
  - Dernières actions terminées
  - Les Principales info à regarder

ROADMAP PATCH 14 JULY

1. PATCH UPTIME-004 — Public Status Page / SLA

C'est le prochain patch prioritaire.

Objectif :

```
/status
/status/<token>
```

Chaque utilisateur pourra publier une page de statut publique ou privée :

```
- statut global : Operational / Degraded / Outage
- liste des sites monitorés
- uptime 24h / 7j / 30j
- incidents ouverts
- incidents résolus récents
- historique visuel 30 jours
- page partageable avec un token
```

C'est ce qui transforme l'uptime en vraie fonctionnalité commerciale.

---

2. PATCH UPTIME-005 — SLA Engine

Objectif :

```
- calcul SLA réel par site
- disponibilité 24h / 7j / 30j / 90j
- downtime cumulé
- nombre d'incidents
- MTTR
- pire incident
- export CSV/PDF
```

Aujourd'hui on affiche déjà des stats, mais il faut les rendre plus solides, plus propres, plus "produit".

3. PATCH DASHBOARD-SEC-001 — Roles Owner/Admin/User

Très important avant ouverture publique.

Objectif :

```
owner Ideo-Lab
admin client
user client
```

Et surtout :

```
- cacher Scheduler aux users normaux
- cacher Cleanup aux users normaux
- cacher Log aux users normaux
- cacher Health technique aux users normaux
- réserver certaines actions au propriétaire
- sécuriser toutes les routes POST
```

Aujourd'hui on a commencé à cloisonner, mais il faut verrouiller proprement.

4. PATCH DASHBOARD-UX-001 — Refonte navigation SaaS

Objectif : rendre l'interface moins "outil admin" et plus "produit client".

Menu cible :

```
Dashboard
New Scan
Jobs
Uptime
Incidents
Status Page
Site Intelligence
Reports
Settings
Logout
```

Et pour owner seulement :

```
Scheduler
Server Health
Logs
Cleanup
Admin
```

5. PATCH INTEL-003 — Site Intelligence depuis Dashboard

Objectif :

```
- lancer Site Intelligence depuis le dashboard
- choisir un site existant
- voir DNS / HTTP / SEO / Performance
- historique par site
- score intelligence
- recommandations
```

Aujourd'hui `intel` marche côté CLI, mais ce n'est pas encore vraiment intégré comme une feature utilisateur.

---

6. PATCH REPORT-001 — Unified Reports Center

Objectif :

```
/reports
```

Regrouper :

```
- rapports de scan sécurité
- rapports uptime
- incidents
- SLA
- site intelligence
- exports PDF/HTML/CSV
```

Le user ne doit pas chercher dans plusieurs pages.

7. PATCH BILLING-001 — License / Plan Limits

Objectif futur :

```
Free / Trial / Pro / Enterprise
```

Limites possibles :

```
- nombre de sites uptime
- fréquence minimale
- nombre de scans sécurité par mois
- accès status page
- accès alertes Slack/WhatsApp/SMS
- conservation historique 30/90/365 jours
```

CE QUI RESTE A FAIRE AU 15/07

Priorité 1 — `DASHBOARD-UX-001` : rendre l'application compréhensible

C'est prioritaire avant d'ajouter encore des modules. L'application est techniquement riche mais produit trop de bruit.

À faire :

```
- menu SaaS propre
- séparation claire :
  Start
  Security Scans
  Uptime
  Intelligence
  Reports
  Settings
- masquer Scheduler / Health / Log / Cleanup aux users normaux
- ajouter help online contextuel
- ajouter un parcours guidé :
  1. lancer un scan
  2. configurer uptime
  3. lire les incidents
  4. corriger / retester
```

Statut : **proposé mais pas encore vraiment intégré dans ton fichier actuel.**

Priorité 2 — `DASHBOARD-SEC-001` : rôles Owner / Admin / User

Avant ouverture publique, c'est indispensable.

À faire :

```
owner Ideo-Lab
admin client
user client
```

Et surtout :

```
- Scheduler visible owner uniquement
- Health / Log / Cleanup owner uniquement
- Uptime Admin accessible user mais limité à ses propres sites
- toutes les routes POST sécurisées par rôle
- éviter qu'un user manipule incident/target d'un autre
```

La roadmap du 14 juillet le listait explicitement comme critique avant ouverture publique. security_guide_roadmap_2026_1

Priorité 3 — `UPTIME-004` : Public Status Page / SLA baseline

C'est la suite naturelle des incidents.

À faire :

```
/status
/status/<token>
```

Avec :

```
- statut global : Operational / Degraded / Outage
- liste des sites monitorés
- uptime 24h / 7j / 30j
- incidents ouverts
- incidents résolus récents
- historique 30 jours
- page partageable avec token
```

C'est ce qui transforme l'uptime en vraie feature commerciale.

Priorité 4 — `UPTIME-005` : SLA Engine solide

Aujourd'hui on affiche déjà des stats, mais il faut les fiabiliser.

À faire :

```
- SLA réel par site
- disponibilité 24h / 7j / 30j / 90j
- downtime cumulé
- nombre d'incidents
- MTTR
- pire incident
- export CSV/PDF
```

Priorité 5 — `INTEL-DASH-003` : Site Intelligence depuis dashboard

Le mode `intel` existe côté orchestrateur, mais il reste trop CLI.

À faire :

```
- bouton Run Site Intelligence depuis dashboard
- choisir un site déjà monitoré
- historique par site
- DNS / HTTP / SEO / Performance en onglets
- score intelligence
- recommandations actionnables
```

Priorité 6 — `INTEL-003` : Reputation / blacklist

Roadmap initiale :

```
- Google Safe Browsing si API key
- VirusTotal si API key
- URLHaus public
- DNSBL basique IP/domain
- statut CLEAN / WARNING / LISTED
```

Très utile commercialement, car un client comprend immédiatement “mon domaine est blacklisté ou non”.

Priorité 7 — `INTEL-004` : Search visibility

À faire :

```
- indexabilité technique sans API
- robots.txt
- sitemap
- canonical
- noindex
- pages crawlables
- option API Bing / Google Custom Search / SerpAPI
- requête marque
- snippets
```

C'est utile, mais moins urgent que SLA/status page.

Priorité 8 — `INTEL-005` : News / press / brand monitoring

À faire plus tard :

```
- RSS
- GDELT
- News API optionnelle
- mentions marque
- sentiment simple
- timeline
```

Priorité 9 — modules qualité web complémentaires

Ils sont dans la roadmap mais doivent venir après le produit central :

```
- accessibilité basique
- contenu / qualité éditoriale
- détection changement / défiguration
- performance avancée Lighthouse / Core Web Vitals
```

La défiguration est probablement la plus utile sécurité :

```
- hash homepage
- title HTML
- taille HTML
- mots-clés suspects
- changement brutal de contenu
- alerte si hacked / casino / viagra / crypto, etc.
```

ORDRE DES PATCHS

1. DASHBOARD-UX-001
   Navigation SaaS + aide online + parcours utilisateur

2. DASHBOARD-SEC-001
   Roles Owner/Admin/User + routes protégées

3. UPTIME-004
   Public Status Page / token partageable

4. UPTIME-005
   SLA Engine solide

5. INTEL-DASH-003
   Site Intelligence lançable depuis le dashboard

6. INTEL-003
   Reputation / blacklist

7. INTEL-004
   Search visibility

8. INTEL-006
   Defacement / change detection

9. REPORT-001
   Unified Reports Center

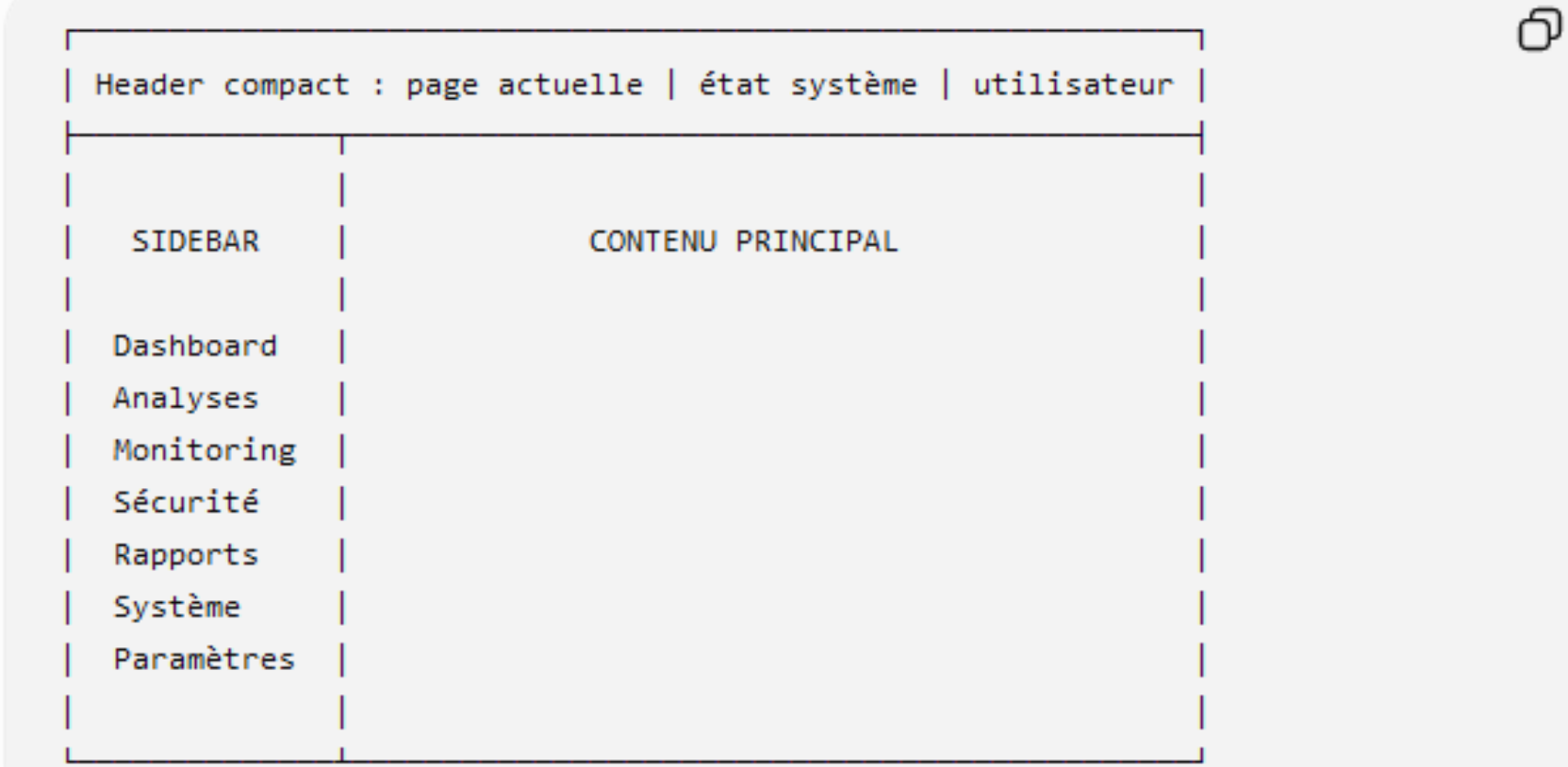
10. BILLING-001
    Plan limits / trial / pro / enterprise

MENU LATERAL

PATCH DASHBOARD-UX-002 — Sidebar SaaS rétractable

La navigation principale doit être déplacée dans une **barre latérale verticale gauche**, permanente sur ordinateur et escamotable sur les petits écrans.

Structure générale retenue



Comportement attendu

Mode étendu

Mode contracté

Seules les icônes restent visibles :

```
[SA]

▣
⌕
◉
◆
▤
⚙
```

Au survol d'une icône, une infobulle indiquera clairement :

```
Analyses
Monitoring
Sécurité
Rapports
Administration
```

Règles ergonomiques

La sidebar devra :

- pouvoir être ouverte ou fermée manuellement avec un bouton ;
- mémoriser le choix de l'utilisateur avec `localStorage` ;
- réduire automatiquement sa largeur lorsque l'écran devient plus petit ;
- devenir un panneau coulissant sur mobile ;
- mettre clairement en évidence la section active ;
- n'ouvrir qu'un seul groupe de sous-menus à la fois ;
- afficher des badges lorsque cela apporte une information utile ;
- conserver les aides contextuelles introduites dans les patchs précédents.

Exemples de badges :

```
Analyses en cours      2
Incidents              1
Rapports disponibles   6
Alertes système        3
```

Ce qui restera horizontal

Il ne faut pas supprimer tous les éléments horizontaux. Ils doivent simplement être réservés à la navigation locale d'une page.

Par exemple, dans la page **Monitoring** :

```
Vue générale | Sites | Incidents | Historique | Configuration
```

Ou dans une fiche d'analyse :

```
Résumé | Vulnérabilités | TLS | Headers | Réseau | Recommandations
```

Ce qui restera horizontal

Il ne faut pas supprimer tous les éléments horizontaux. Ils doivent simplement être réservés à la navigation locale d'une page.

Par exemple, dans la page **Monitoring** :

```
Vue générale | Sites | Incidents | Historique | Configuration
```

Ou dans une fiche d'analyse :

```
Résumé | Vulnérabilités | TLS | Headers | Réseau | Recommandations
```

La règle devient donc simple :

- **sidebar gauche** : navigation entre les grandes fonctionnalités ;
- **onglets horizontaux** : navigation interne à la fonctionnalité sélectionnée ;
- **boutons** : actions immédiates ;
- **menus déroulants** : options secondaires.

Header supérieur simplifié

Le haut de l'écran ne doit plus contenir toute la navigation. Il pourra uniquement contenir :

```
[☰]  Analyses / Résultat du scan

Serveur : opérationnel
Orchestrateur : inactif
Dernière analyse : il y a 12 min

[Thème] [Aide] [Notifications] [Utilisateur]
```

Le bouton ☰ permettra de contracter ou d'ouvrir la sidebar.

Organisation fonctionnelle proposée

1. Vue d'ensemble

- tableau de bord général ;
- score global ;
- analyses récentes ;
- incidents ;
- état de l'orchestrateur ;
- état des services ;
- disponibilité des sites.

2. Analyses

- lancer une analyse ;
- analyses en cours ;
- résultats ;
- historique ;
- comparaison entre deux runs ;
- analyses planifiées.

3. Monitoring

- supervision uptime ;
- sites surveillés ;
- incidents ;
- temps de réponse ;
- historique de disponibilité ;
- configuration des fréquences.

4. Sécurité

- vulnérabilités ;
- TLS et certificats ;
- headers HTTP ;
- ports ;
- DNS ;
- technologies détectées ;
- recommandations.

5. Rapports

- génération de rapports ;
- rapports HTML ;
- exports PDF ou JSON ;
- historique ;
- partage ;
- planification.

6. Système

- état de l'orchestrateur ;
- services Linux ;
- daemons ;
- tâches planifiées ;
- files d'attente ;
- journaux ;
- diagnostics serveur.

Organisation fonctionnelle proposée 7

7. Administration

- utilisateurs ;
- rôles ;
- abonnements ou quotas ;
- paramètres globaux ;
- configuration SMTP ;
- sécurité applicative.

Aspect visuel

Pour rester cohérent avec ce que tu demandes depuis plusieurs patchs :

- bleu principal conservé ;
- fond de sidebar légèrement plus soutenu que le contenu ;
- bordures visibles ;
- relief léger ;
- icônes lisibles ;
- états actifs très distincts ;
- contraste renforcé ;
- thème clair et thème sombre compatibles ;
- animations courtes et discrètes ;
- pas d'effets gadgets.

Exemple de section active :

```
| ◆  Sécurité          |
```

Découpage du patch

PATCH DASHBOARD-UX-002A — Structure de la sidebar

- conteneur général ;
- sidebar gauche ;
- zone de contenu ;
- header compact ;
- bouton de contraction ;
- mémorisation de l'état.

PATCH DASHBOARD-UX-002B — Navigation et groupes repliables

- catégories ;
- sous-menus ;
- section active ;
- ouverture exclusive d'un groupe ;
- infobulles du mode contracté.

PATCH DASHBOARD-UX-002C — Responsive mobile

- sidebar coulissante ;
- overlay ;
- fermeture tactile ;
- adaptation tablette et mobile.

PATCH DASHBOARD-UX-002D — Nettoyage de l'ancienne navigation

- suppression des doublons horizontaux ;
- conservation uniquement des onglets contextuels ;
- déplacement des boutons au bon niveau ;
- harmonisation des pages existantes.

Dernier Patch

PATCH DASHBOARD-UX-002E — Finitions SaaS

- badges ;
- état des services ;
- notifications ;
- thème clair et sombre ;
- animations ;
- accessibilité clavier ;
- aides contextuelles.

La prochaine intervention logique est donc :

> **PATCH DASHBOARD-UX-002A — Shell SaaS avec sidebar gauche rétractable**

BILAN DEVELOPPEMENT

BILAN FEATURES

- 1. Anomalie immédiate après la restauration
- 2. État de la roadmap prioritaire des pages 26 à 32
- 3. Roadmap Intelligence des pages 18 à 22
- 4. To-do Uptime de la page 24
- 5. Éléments du plan initial de l’orchestrateur
- 6. Fonctions longues encore absentes de la première partie de la roadmap
- 7. Ordre de reprise à respecter

2. État de la roadmap prioritaire

État de la roadmap prioritaire

- **UPTIME-004** — Public Status Page / SLA
- UPTIME-005 — SLA Engine
- DASHBOARD-SEC-001 — Rôles Owner
- DASHBOARD-UX-001 — Navigation SaaS
- Site Intelligence depuis le Dashboard
- REPORT-001 — Unified Reports Center
- BILLING-001 — License / Plan Limits

UPTIME-004 — Public Status Page / SLA

État : totalement absent.

La roadmap prévoit :

```
/status
/status/<token>
```

Avec :

- statut global Operational / Degraded / Outage ;
- liste des sites publiés ;
- uptime 24 h, 7 j et 30 j ;
- incidents ouverts ;
- incidents résolus récents ;
- historique visuel ;
- page partageable par token.

Aucune route `/status`, aucune table de configuration de page publique et aucune fonction de rendu correspondante n'existent actuellement. C'est le prochain gros patch indiqué comme prioritaire dans la roadmap. security_guide_roadmap_2026_1

UPTIME-005 — SLA Engine

État : partiellement développé.

Déjà présent :

```
uptime 24 h
uptime 7 jours
uptime 30 jours
downtime cumulé
nombre d'incidents
historique des probes
historique des incidents
```

Encore manquant :

```
uptime 90 jours
MTTR
pire incident
SLA contractuel par site
objectif SLA, par exemple 99,9 %
écart par rapport à l'objectif
export CSV
export PDF
rapport SLA propre
```

Il n'existe pas de route dédiée telle que :

```
/sla
/sla-export
```

DASHBOARD-SEC-001 — Rôles Owner / Admin / User

État : partiellement développé.

Déjà présent :

- authentification des utilisateurs ;
- isolation des bases utilisateur ;
- propriétaire reconnu par liste d'adresses email ;
- protection de `/scheduler` ;
- protection maintenant ajoutée à `/health`, `/server-log` et `/cleanup` ;
- protections owner des actions POST du Scheduler.

Encore manquant :

1. Aucun champ `role` dans `dashboard_user`.

Le modèle actuel ne distingue que :

```
utilisateur actif
utilisateur propriétaire reconnu par email
```

Il manque les vrais rôles :

```
owner
admin
user
```

2. Le groupe **Administration** est affiché à tous les utilisateurs.

Les routes sont interdites aux utilisateurs normaux, mais les boutons restent visibles. La roadmap demande de les cacher complètement.

DASHBOARD-UX-001 — Navigation SaaS

État : partiellement développé.

Déjà présent :

- sidebar gauche ;
- sidebar rétractable ;
- groupes de navigation ;
- mode mobile coulissant ;
- topbar ;
- thème clair et sombre ;
- tooltips lorsque la sidebar est repliée ;
- restauration des anciens menus techniques.

Encore manquant par rapport au menu cible de la page 29 :

```
Status Page
Reports
Settings
Admin
```

Autres éléments inachevés :

- le groupe Administration est visible aux utilisateurs normaux ;
- le groupe Account est actuellement dupliqué ;
- l'ancien menu horizontal `nav()` est encore affiché dans pratiquement toutes les pages ;
- on a donc deux systèmes de navigation simultanés ;
- la fonction `nav()` n'est plus synchronisée avec la sidebar ;
- `Targets`, `Scheduler`, `Health`, `Log` et `Cleanup` ne sont pas tous présents dans

Site Intelligence depuis le Dashboard

État : partiellement développé.

L'orchestrateur possède bien le mode CLI :

```bash
python3 ideolab_security_orchestrator.py intel ...
```

Il exécute quatre domaines :

```
intel-dns
intel-http
intel-seo
intel-performance
```

Le dashboard permet déjà :

- d'afficher les runs Intel ;
- d'ouvrir un run ;
- d'afficher les métriques ;
- d'afficher les findings ;
- d'afficher les résultats des outils ;
- d'afficher le score et la décision.

Mais il manque toujours :

```
lancer Site Intelligence depuis le dashboard
choisir un site existant
suivre le job en direct
historique filtré par site
relancer un ancien site
afficher clairement DNS / HTTP / SEO / Performance par onglets
```

REPORT-001 — Unified Reports Center

État : totalement absent du dashboard.

L'orchestrateur génère déjà :

```
PDF
HTML
TXT
rapport email
```

Mais le dashboard n'a pas :

```
/reports
liste centralisée des rapports
téléchargement sécurisé
filtre par type
filtre par site
filtre par utilisateur
rapports Security
rapports Uptime
rapports Incidents
rapports SLA
rapports Site Intelligence
exports CSV
```

Les fichiers sont produits sur disque, mais aucun centre de rapports utilisateur n'existe.

BILLING-001 — License / Plan Limits

État : fondation de licence présente, limites produit absentes.

Déjà présent :

- demande de licence ;
- activation en ligne ;
- cache local ;
- fingerprint machine ;
- statut VALID / GRACE ;
- expiration ;
- période hors ligne ;
- édition Trial / Pro / Enterprise ;
- licence vérifiée par le dashboard et l'orchestrateur.

Encore manquant :

```
page Plan / License
affichage du plan courant
limite du nombre de sites Uptime
limite du nombre de scans mensuels
fréquence minimale autorisée
limite de rétention
activation selon plan de Slack / SMS / WhatsApp
accès à la Status Page
accès aux exports
compteurs mensuels
blocage effectif en cas de dépassement
```

Le champ `features` est reçu et stocké dans le cache de licence, mais il n'est pratiquement jamais utilisé pour autoriser ou refuser une fonctionnalité.

3. Roadmap Intelligence

INTEL-003 — Reputation / blacklist

Il manque notamment :

```
Google Safe Browsing
VirusTotal
URLhaus
PhishTank
Spamhaus / DNSBL
OpenPhish
abuse.ch
statut CLEAN / WARNING / LISTED
```

Aucune fonction, table ou route dédiée n'existe.

INTEL-004 — Search visibility

Une partie de l'indexabilité technique existe déjà dans `intel-seo` :

```
robots.txt
sitemap
canonical
noindex
title
meta description
```

Mais il manque :

```
requête de marque
pages réellement trouvées
snippets
pages indexées
Bing API
Google Custom Search
SerpAPI
historique de visibilité
```

INTEL-005 — News / press / brand monitoring

Totalement absent :

```
RSS
GDELT
News API
mentions de marque
sentiment simple
timeline
fréquence de publication
signal de crise
```

Roadmap immédiate

```
SCAN-JOB-001A
Background scan + terminal dynamique
→ validé

SCAN-JOB-001B
Restart-safe jobs + récupération automatique
→ prochain patch

SCAN-JOB-001C
Cancel / Stop scan propre et contrôlé

SCAN-JOB-001D
File d’attente, limite de concurrence et prévention des doubles scans
```

- les niveaux de priorité complexes ;
- la réorganisation manuelle de la file ;
- le drag-and-drop ;
- les réservations de ressources CPU/RAM ;
- les calendriers de scans ;
- les quotas mensuels ;
- les files Redis ou Celery ;
- les workers multiples ;
- les statistiques avancées de temps d'attente ;
- tout nouveau daemon ou service systemd.

Le module sera considéré terminé lorsque ces six scénarios passent :

```
1. Un scan lancé continue après fermeture du navigateur.

2. Un scan continue après redémarrage du Dashboard.

3. Pause, Resume et Cancel fonctionnent.

4. Deux clics sur la même cible ne créent pas deux scans.

5. Un scan excédentaire reste queued sans PID.

6. Quand un scan termine, le premier queued démarre automatiquement.
```

Une fois ces six tests validés, nous ne toucherons plus au moteur des jobs.

Plan fermé de DASHBOARD-SEC-001

DASHBOARD-SEC-001A — Role Schema & Permission Core

Ce bloc ajoutera :

- la colonne SQL `role` dans `dashboard_user` ;
- les rôles stricts `owner`, `admin`, `user` ;
- `user` comme rôle obligatoire de toute nouvelle inscription publique ;
- la migration automatique des comptes existants ;
- la promotion automatique de `gon2000fr@gmail.com` en `owner` ;
- des fonctions centrales `has_role()`, `require_role()` et `route_is_allowed()` ;
- une matrice unique pour toutes les routes GET et POST ;
- la suppression progressive de la reconnaissance du propriétaire uniquement par e-mail ;
- le contrôle systématique de l'appartenance des jobs, runs, incidents et configurations au bon utilisateur.

DASHBOARD-SEC-001B — Owner Administration Center

Ce dernier bloc ajoutera :

- la page `/admin/users` ;
- la liste des utilisateurs inscrits ;
- le rôle, l'état actif, la dernière connexion et le nombre de sessions ;
- l'activation et la désactivation d'un compte ;
- le changement de rôle, réservé à l'owner ;
- le groupe Administration visible uniquement aux personnes autorisées ;
- Scheduler, Health, Log et Cleanup invisibles et interdits aux autres rôles ;
- une connexion de support contrôlée au compte d'un utilisateur ;
- un bandeau permanent pendant cette session de support ;
- une action explicite pour revenir au compte owner ;
- une trace SQL de toute impersonation et de toute modification de rôle.

DASHBOARD-SEC-001-VALIDATION

Le module sera gelé lorsque ces scénarios passeront :

1. Toute nouvelle inscription reçoit le rôle user.

2. Le compte owner est stocké comme owner en SQL.

3. Un user ne voit pas le groupe Administration.

4. Un accès direct user à Scheduler, Health, Log ou Cleanup retourne 403.

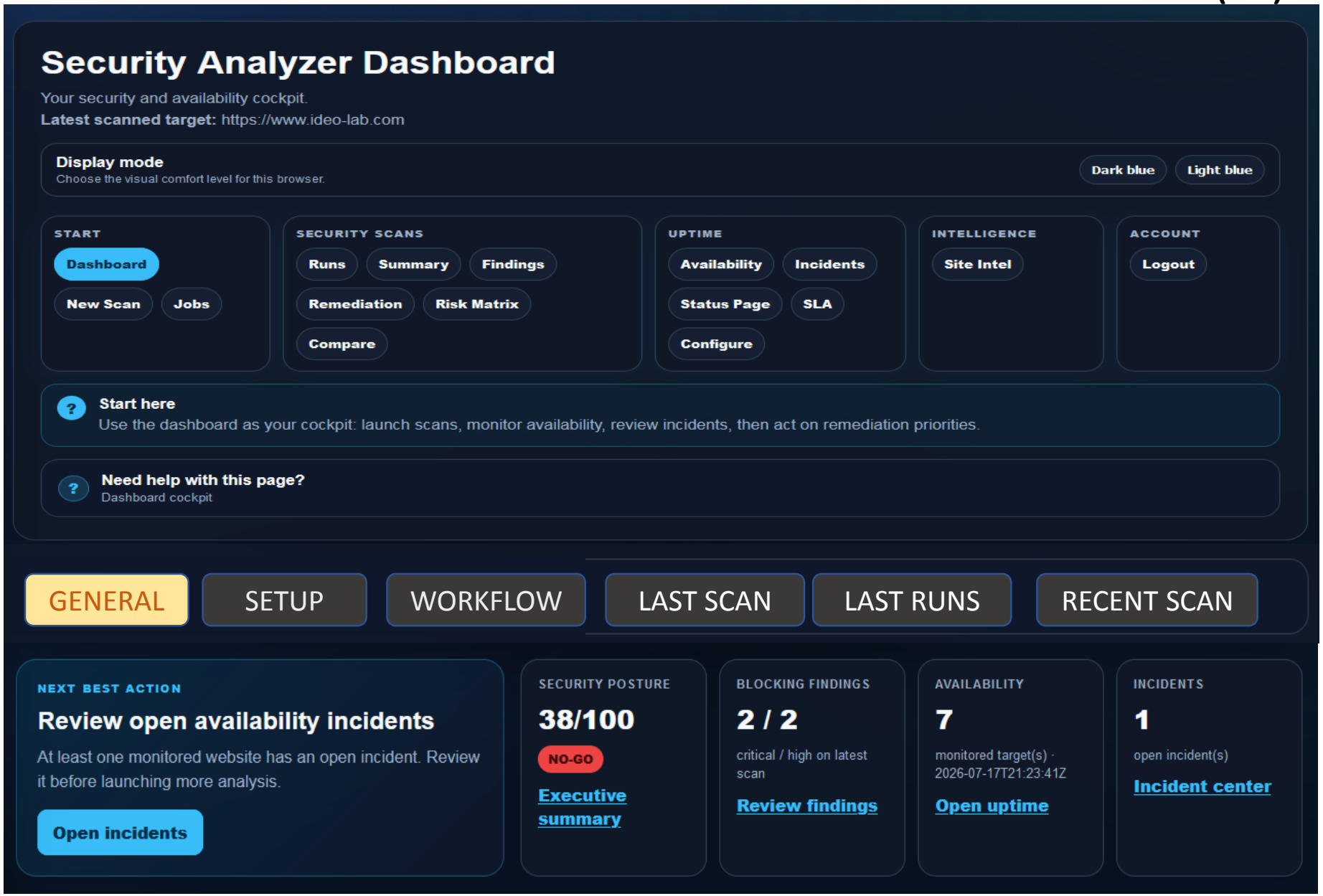
5. Les routes POST sont protégées avec la même politique que les routes GET.

6. Un utilisateur ne peut jamais lire ou modifier les données d’un autre.

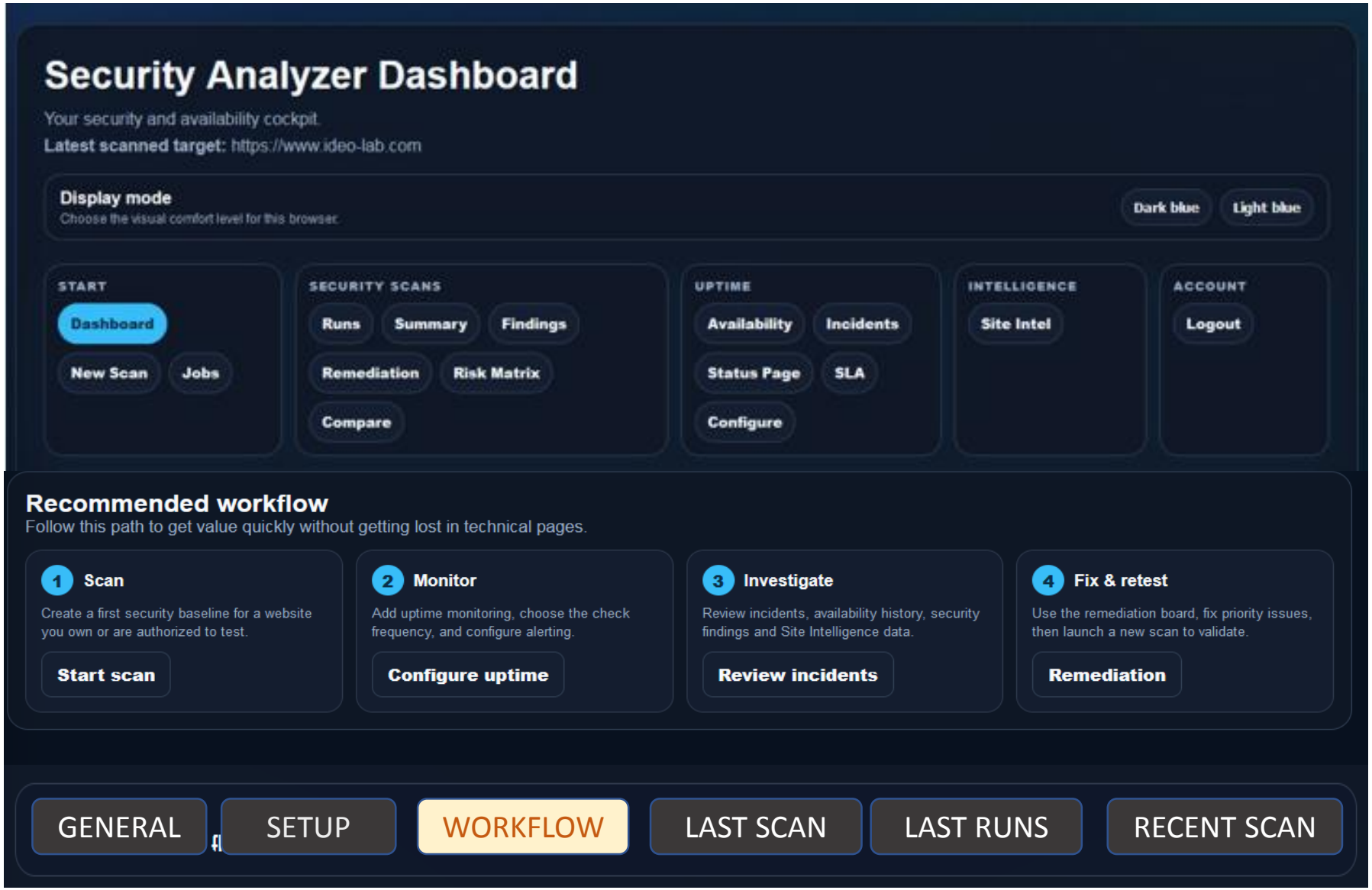
7. L’owner peut administrer les comptes et les rôles.

8. Toute session de support est visible, limitée et auditée.

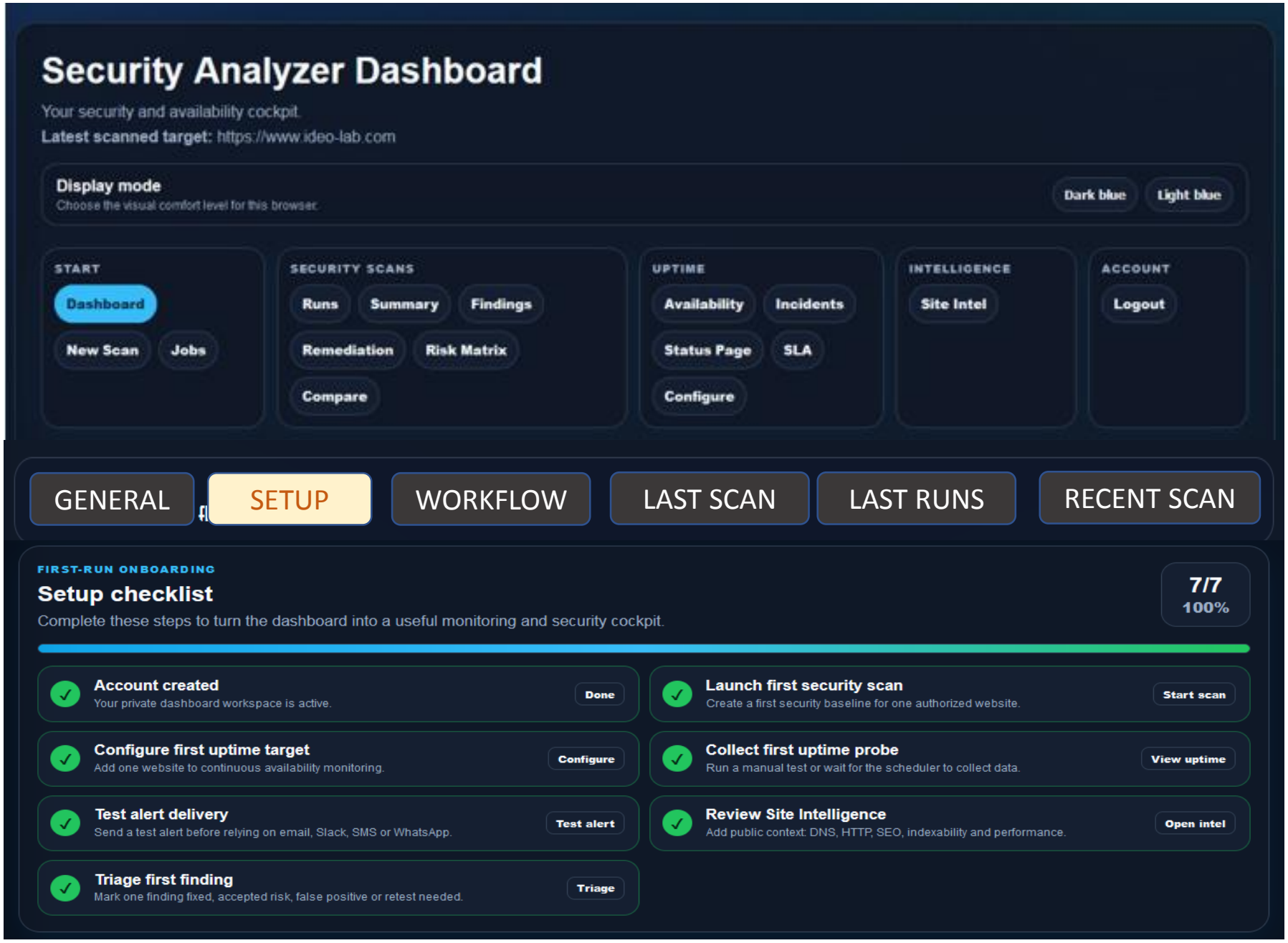
DASHBOARD MAIN PAGE - GENERAL (1)



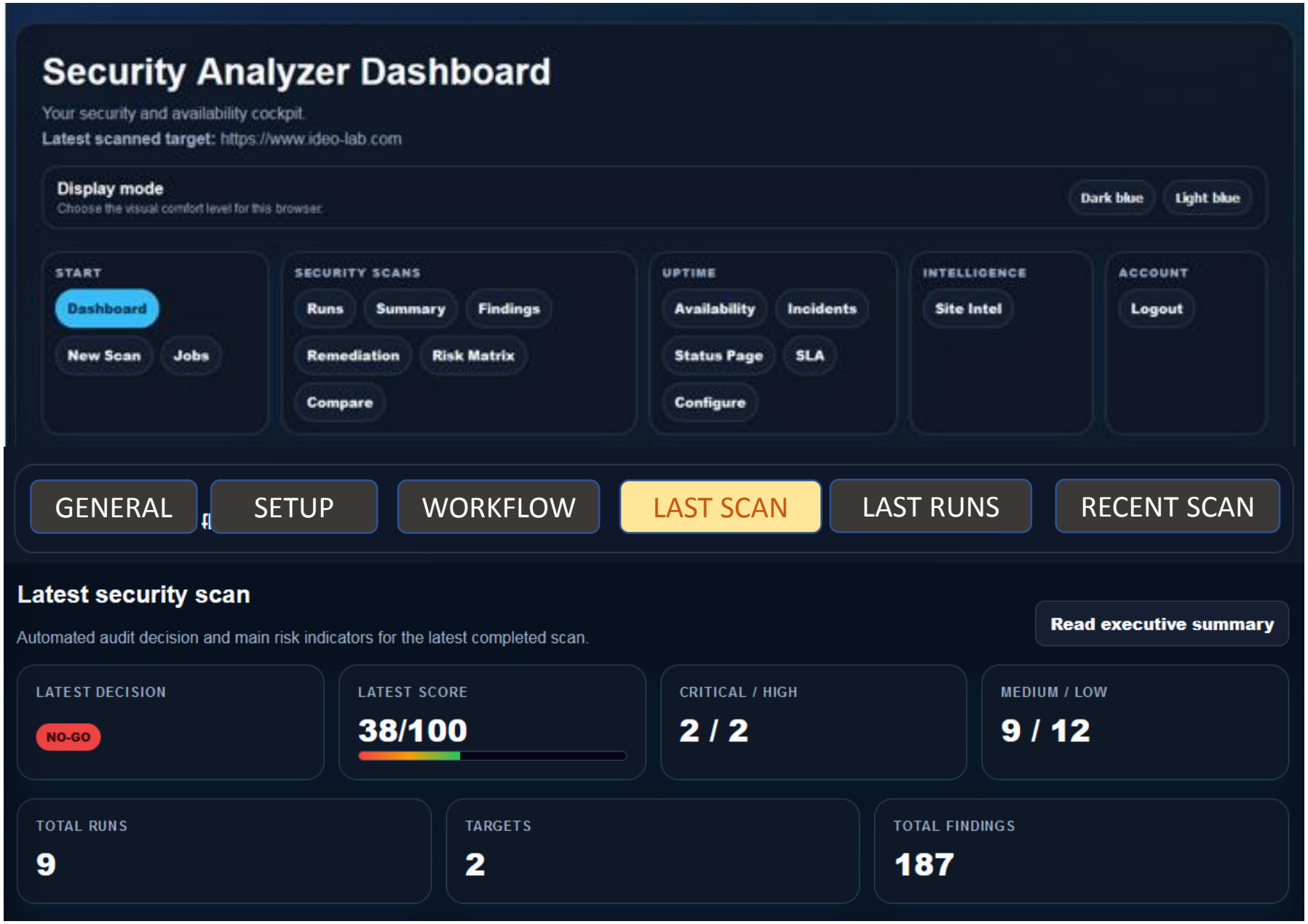
WORKFLOW PARAGRAPH



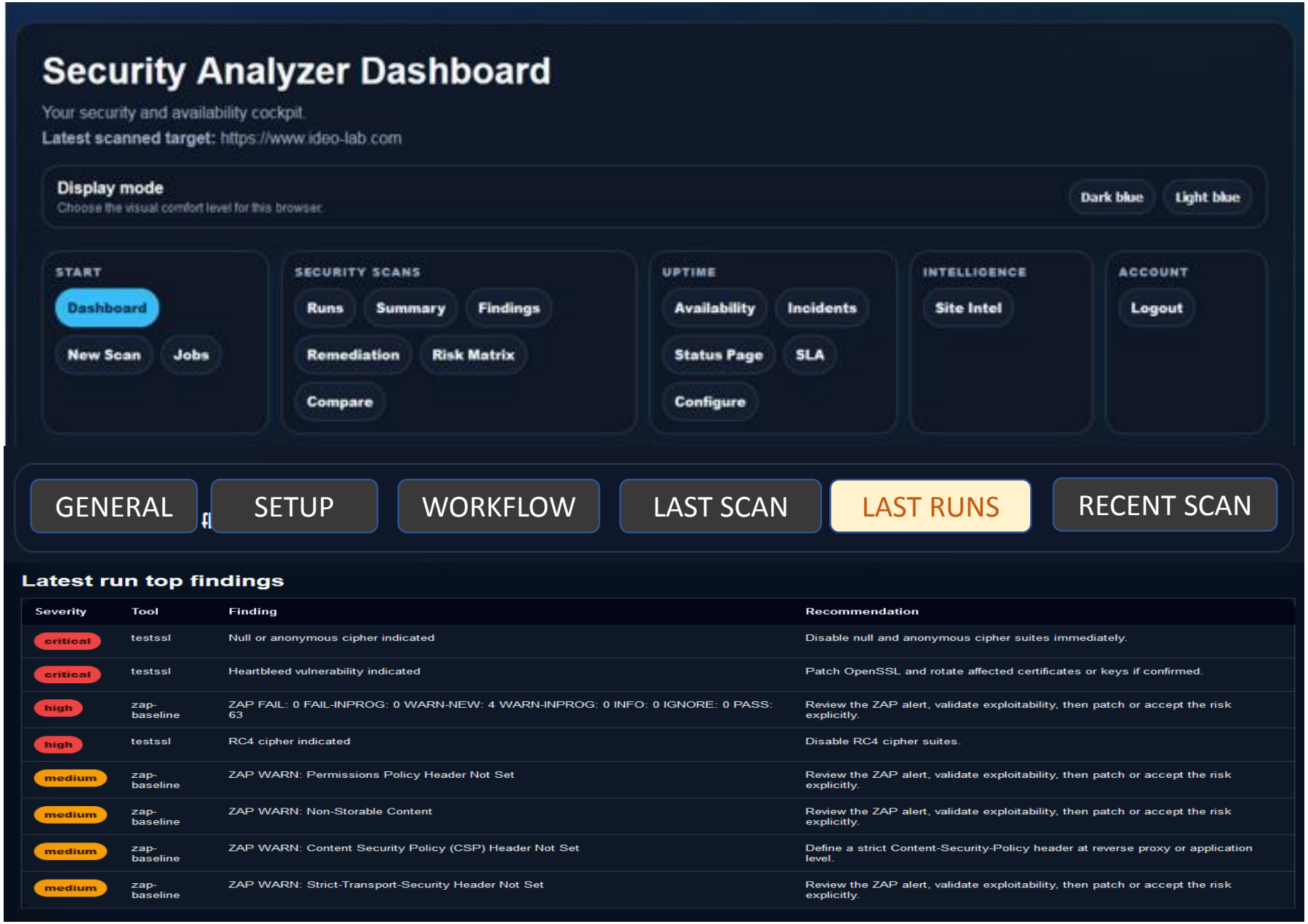
SETUP PARAGRAPH



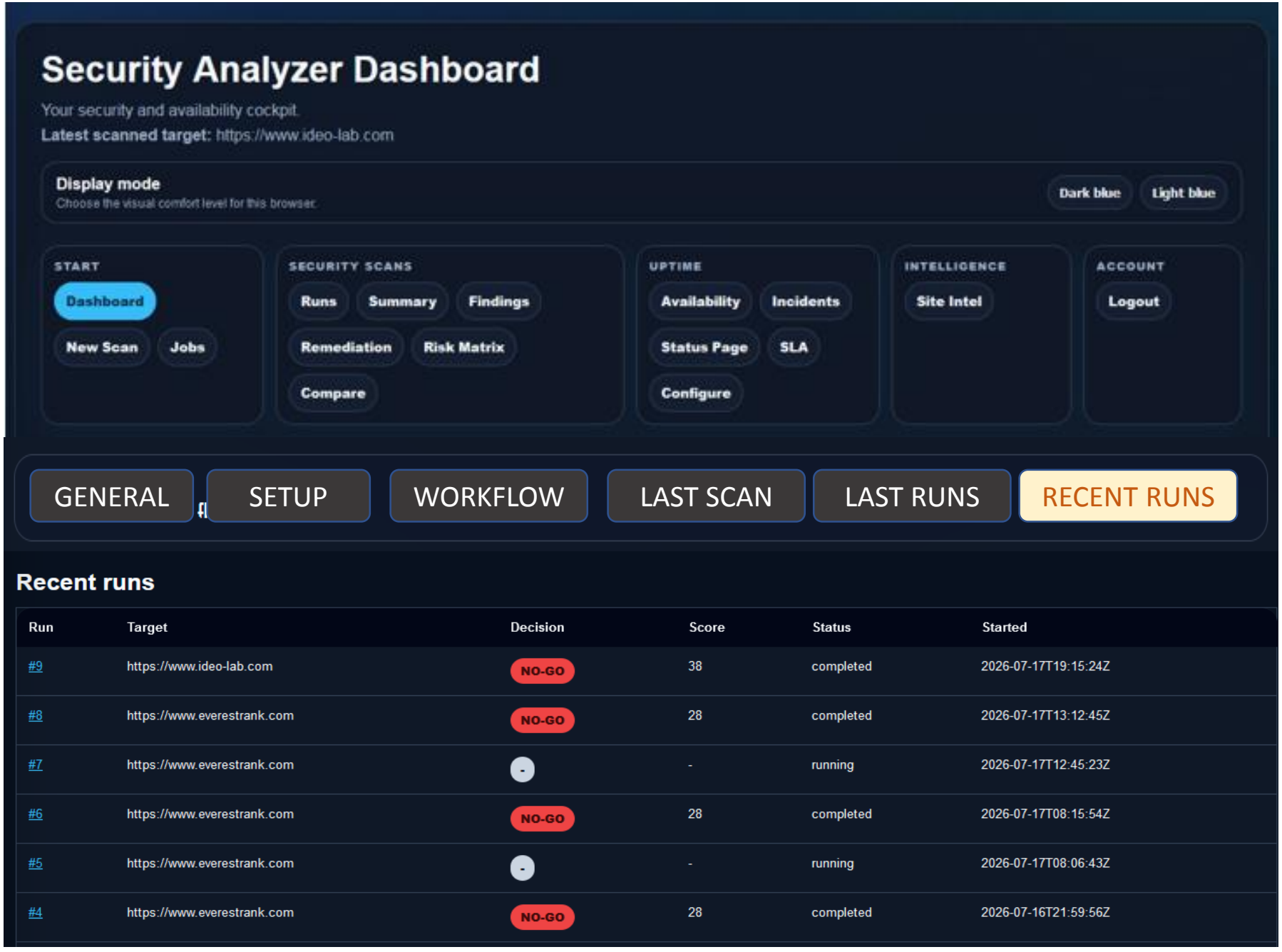
LATEST SCAN


Security Analyzer Dashboard
Your security and availability cockpit.
Latest scanned target: https://www.ideo-lab.com
Display mode
Choose the visual comfort level for this browser.
Dark blue
Light blue
START
Dashboard
New Scan
Jobs
SECURITY SCANS
Runs
Summary
Findings
Remediation
Risk Matrix
Compare
UPTIME
Availability
Incidents
Status Page
SLA
Configure
INTELLIGENCE
Site Intel
ACCOUNT
Logout
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT SCAN
Latest security scan
Automated audit decision and main risk indicators for the latest completed scan.
Read executive summary
LATEST DECISION
NO-GO
LATEST SCORE
38/100
CRITICAL / HIGH
2 / 2
MEDIUM / LOW
9 / 12
TOTAL RUNS
9
TARGETS
2
TOTAL FINDINGS
187

LAST RUNS


Security Analyzer Dashboard
Your security and availability cockpit.
Latest scanned target: https://www.ideo-lab.com
Display mode
Choose the visual comfort level for this browser.
Dark blue
Light blue
START
Dashboard
New Scan
Jobs
SECURITY SCANS
Runs
Summary
Findings
Remediation
Risk Matrix
Compare
UPTIME
Availability
Incidents
Status Page
SLA
Configure
INTELLIGENCE
Site Intel
ACCOUNT
Logout
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT SCAN
Latest run top findings
Severity
Tool
Finding
Recommendation
critical
testssl
Null or anonymous cipher indicated
Disable null and anonymous cipher suites immediately.
critical
testssl
Heartbleed vulnerability indicated
Patch OpenSSL and rotate affected certificates or keys if confirmed.
high
zap-baseline
ZAP FAIL: 0 FAIL-INPROG: 0 WARN-NEW: 4 WARN-INPROG: 0 INFO: 0 IGNORE: 0 PASS: 63
Review the ZAP alert, validate exploitability, then patch or accept the risk explicitly.
high
testssl
RC4 cipher indicated
Disable RC4 cipher suites.
medium
zap-baseline
ZAP WARN: Permissions Policy Header Not Set
Review the ZAP alert, validate exploitability, then patch or accept the risk explicitly.
medium
zap-baseline
ZAP WARN: Non-Storable Content
Review the ZAP alert, validate exploitability, then patch or accept the risk explicitly.
medium
zap-baseline
ZAP WARN: Content Security Policy (CSP) Header Not Set
Define a strict Content-Security-Policy header at reverse proxy or application level.
medium
zap-baseline
ZAP WARN: Strict-Transport-Security Header Not Set
Review the ZAP alert, validate exploitability, then patch or accept the risk explicitly.

Recent runs


Security Analyzer Dashboard
Your security and availability cockpit.
Latest scanned target: https://www.ideo-lab.com
Display mode
Choose the visual comfort level for this browser.
Dark blue
Light blue
START
Dashboard
New Scan
Jobs
SECURITY SCANS
Runs
Summary
Findings
Remediation
Risk Matrix
Compare
UPTIME
Availability
Incidents
Status Page
SLA
Configure
INTELLIGENCE
Site Intel
ACCOUNT
Logout
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT RUNS
Recent runs
Run
Target
Decision
Score
Status
Started
#9
https://www.ideo-lab.com
NO-GO
38
completed
2026-07-17T19:15:24Z
#8
https://www.everestrank.com
NO-GO
28
completed
2026-07-17T13:12:45Z
#7
https://www.everestrank.com
-
-
running
2026-07-17T12:45:23Z
#6
https://www.everestrank.com
NO-GO
28
completed
2026-07-17T08:15:54Z
#5
https://www.everestrank.com
-
-
running
2026-07-17T08:06:43Z
#4
https://www.everestrank.com
NO-GO
28
completed
2026-07-16T21:59:56Z

SECURITY GUIDE BY IDEO-LAB

18 JULY 2026

AUTEUR : GUILLAUME ONEILL

v 1.8

Fonctions connexes très utiles

1. Disponibilité / continuité de service

```
- site online/offline
- code HTTP retourné
- temps de réponse
- certificat encore valide
- redirection HTTP -> HTTPS OK
- contenu minimum présent
- variation du titre HTML
- variation hash de la homepage
- erreurs 5xx
- erreurs DNS
- timeout
```

Périodes :

```
- 24 heures
- 7 jours
- 30 jours
- 90 jours
```

Résultat dashboard :

```
Uptime 24h     : 100%
Uptime 7j      : 99.82%
Uptime 30j     : 99.64%
Incidents      : 3
Temps moyen    : 284 ms
Temps max      : 2.8 s
Dernière panne : 2026-07-10 03:14
```

Ça nécessite un mini scheduler côté dashboard ou un service dédié.

2. DNS / domaine / infrastructure publique

Analyse totalement distante :

```
- A / AAAA
- CNAME
- MX
- TXT
- SPF
- DKIM probable
- DMARC
- CAA
- NS
- TTL
- DNSSEC
- WHOIS / RDAP si disponible
- date d'expiration domaine
- registrar
- nameservers
- IP publiques
- ASN / hébergeur
- géolocalisation approximative IP
```

Très utile pour un rapport client.

Exemples de diagnostics :

```
DMARC absent
SPF trop permissif
CAA absent
TTL trop bas ou incohérent
Domaine proche expiration
MX exposé sans politique DMARC
```

3. SEO technique

```
- title présent / longueur correcte
- meta description présente
- canonical
- robots.txt
- sitemap.xml
- hreflang
- balises H1/H2
- OpenGraph
- Twitter Card
- schema.org JSON-LD
- viewport mobile
- alt images
- liens cassés internes
- profondeur des liens
- pages indexables / noindex
- redirections multiples
- 404 importantes
```

Score possible :

```
SEO technical score : 72/100
```

Avec recommandations :

```
- Ajouter meta description
- Corriger canonical
- Ajouter sitemap.xml
- Réduire les redirections
- Ajouter OpenGraph image
```

4. Visibilité moteurs de recherche

```
Mode sans API :
- robots.txt
- sitemap.xml
- balises noindex/nofollow
- canonical
- présence sitemap
- pages déclarées dans sitemap
- pages accessibles
- structure SEO

Mode avec API optionnelle :
- Bing Web Search API
- Google Custom Search API
- SerpAPI / DataForSEO si le client fournit une clé
```

Données utiles :

```
- nombre approximatif de pages indexées
- requêtes de marque
- présence homepage sur requête marque
- titres visibles dans SERP
- snippets
- pages mal indexées
```

Dans le dashboard :

```
Search visibility
    Brand query visible : YES
    Homepage found      : YES
    Indexed sample      : 42 pages
    Sitemap URLs        : 128
    Noindex pages       : 3
```

5. Réputation sécurité / blacklist

Analyse distante très utile :

```
- Google Safe Browsing API
- VirusTotal URL/domain API
- URLhaus
- PhishTank
- Spamhaus / DNSBL
- OpenPhish
- abuse.ch
```

Score :

```
Reputation status : CLEAN / WARNING / LISTED
```

Findings :

```
- Domaine listé dans une blacklist
- IP signalée comme abusive
- URL détectée dans une base phishing
```

Ça donne énormément de valeur au produit.

6. Presse / actualités / marque

Sources possibles :

```
- RSS public du site
- flux news configurés
- Bing News API
- GDELT
- Google News via source externe/API
- recherche web API
```

Objectif :

```
- mentions récentes de la marque
- sentiment approximatif
- sujets associés
- signaux de crise
- articles récents
- fréquence de publication
```

Dashboard :

```
Brand / News monitoring
    Mentions last 7d  : 12
    Positive          : 5
    Neutral           : 6
    Negative          : 1
    Last mention      : ...
```

7. Performance web

Sans être un clone de Lighthouse, on peut déjà faire :

```
- temps DNS
- temps connect
- temps TLS
- time to first byte
- taille HTML
- taille homepage
- nombre assets CSS/JS/images
- compression gzip/brotli
- cache-control
- CDN apparent
- images lourdes
```

Puis en niveau 2 :

```
- Lighthouse CLI
- Core Web Vitals via PageSpeed API
- mobile score
- accessibility score
- best practices score
```

8. Accessibilité / conformité basique

Analyse distante HTML :

```
- images sans alt
- boutons/liens sans texte
- absence lang HTML
- contrastes difficiles à vérifier sans navigateur, mais possible plus tard
- titres hiérarchiques incohérents
- formulaires sans label
```

Très utile pour agences web / clients corporate.

9. Analyse contenu / qualité éditoriale

9. Analyse contenu / qualité éditoriale

À distance :

```
- langue détectée
- longueur texte homepage
- densité contenu réel vs HTML
- duplicate title
- duplicate meta description
- pages vides
- pages pauvres
- textes trop courts
- présence mentions légales
- privacy policy
- terms
- contact page
```

10. Détection changement / défiguration

Très utile sécurité + supervision :

```
- hash homepage
- titre HTML
- taille HTML
- mots clés attendus
- capture textuelle
- changement brutal de contenu
- apparition mots suspects : hacked, casino, viagra, bitcoin, etc.
```

Cela donne un module :

```
Defacement Watch
```

Organisation produit proposée

Dans le dashboard, on pourrait avoir à terme :

```
New Scan
Jobs
Dashboard
Runs
Findings
Remediation
Diff
Health
Log
Cleanup

+ Site Intelligence
+ Uptime
+ SEO
+ Search Visibility
+ DNS / Domain
+ Reputation
+ News / Mentions
```

Mais côté code, il ne faut pas tout mélanger dans `audit`.

Roadmap propre - Patchs

TO DO LIST - july 2026

A FAIRE ET A PREVOIR - 2

- Il faut ajouter une clause About : Header page
- Il faut ajouter une clause contact : Header page
- Il faut ajouter une clause Privacy : Footer Page

- Il faut ajouter un footer :
  - contact email : contact@ideo-lab.com
  - Présentation société (Lockheed LLC & Ideo-lab)
  - Adresse

- Une procédure d'auto-test qui permettra de valider que l'ensemble du produit « Security analyzer » fonctionne parfaitement

A FAIRE ET A PREVOIR - 3

- Mettre dans le menu du header un menu (genre notifications, rappel) : (j'aime bien l'icone « cloche » avec en rouge en haut à droite de l'icone, un petit chiffre rouge (du nombre de notifications récentes non lues ou reardées)
  - Dernières actions terminées
  - Les Principales info à regarder

- **Alertes par** :
  - Emails
  - WhatsApp
  - SMS

REALISES ET RESTE A FAIRE

F. SLA Engine

La page 24 du PDF indiquait comme manquants la route `/sla`, les 90 jours, le MTTR, le pire incident, l'objectif contractuel et l'écart. Ces éléments sont maintenant présents :

- route `/sla` ;
- périodes 24 heures, 7 jours, 30 jours et 90 jours ;
- objectif SLA configurable par site ;
- disponibilité réellement mesurée ;
- écart par rapport à l'objectif ;
- statut `met`, `breached` ou `no data` ;
- temps d'indisponibilité ;
- nombre d'incidents ;
- incidents ouverts ;
- MTTR ;
- pire incident ;
- détail par site ;
- exclusion des sondes et incidents de test.

Il reste seulement :

- export CSV ;
- export PDF ;
- génération d'un véritable fichier de rapport SLA.

`UPTIME-005` est donc **fonctionnel mais non totalement fermé.**

DASHBOARD-SEC-001B — Encore absent

La page 36 prévoit un véritable centre d'administration qui n'existe pas encore :

- route `/admin/users` ;
- liste des comptes ;
- état actif/inactif ;
- dernière connexion ;
- nombre de sessions ;
- activation et désactivation d'un compte ;
- changement de rôle ;
- journal des changements ;
- session de support contrôlée ;
- bandeau d'impersonation ;
- retour explicite au compte owner ;
- audit des impersonations.

Sur les huit critères de la page 37, les six premiers sont largement couverts. Les critères suivants restent ouverts :

1. l'owner peut administrer les comptes et les rôles ;
2. toute session de support est visible, limitée et auditée.

4. Navigation et expérience utilisateur

Ce qui est réalisé

`DASHBOARD-UX-001` a beaucoup progressé :

- sidebar SaaS ;
- sidebar rétractable ;
- groupes de navigation ;
- navigation mobile coulissante ;
- topbar ;
- thèmes clair et sombre ;
- tooltips en mode rétracté ;
- groupe Administration filtré par rôle ;
- Help et FAQ ;
- identité du compte connecté ;
- Dashboard principal organisé par six onglets horizontaux ;
- responsive desktop, tablette et mobile.

Le Dashboard principal ne présente plus tous ses blocs verticalement en même temps.

Ce qui reste imparfait

L'ancien composant `nav()` est encore affiché dans la majorité des pages. Cela entraîne toujours deux niveaux de navigation :

```
Sidebar SaaS
+
ancien menu technique dans le contenu
```

Ce n'est pas bloquant, et il ne faut pas le supprimer brutalement. Il faudra, page par page, décider s'il reste utile ou s'il peut être remplacé par une navigation contextuelle plus légère.

Les entrées suivantes prévues par le cahier sont également absentes :

- Reports ;
- Settings ;
- Admin Users.

Le Dashboard possède maintenant `Status Page`, ce qui n'était pas encore le cas lors de la rédaction de la page 26.

5. Site Intelligence

Déjà réalisé

Le Dashboard sait :

- lire les runs Intel ;
- ouvrir un run ;
- afficher le score et la décision ;
- afficher les findings ;
- afficher les résultats des outils ;
- afficher les métriques ;
- exploiter les tables DNS, HTTP, SEO et performance produites par l'orchestrateur.

Toujours manquant

La page `/intel` demande encore à l'utilisateur d'exécuter une commande dans un terminal :

```bash
python3 ideolab_security_orchestrator.py intel ...
```

Il manque donc exactement ce que décrit la page 27 :

- formulaire de lancement depuis le Dashboard ;
- sélection d'un site existant ;
- saisie d'un nouveau site ;
- confirmation d'autorisation ;
- job Intel en arrière-plan ;
- progression et terminal live ;
- historique filtré par site ;

6. Reports Center

`REPORT-001` est toujours absent, conformément à la page 28 :

- aucune route `/reports` ;
- aucune liste centralisée ;
- aucun téléchargement sécurisé ;
- aucun filtre par type ou site ;
- aucun filtre par utilisateur pour l'owner ;
- aucun rapport consolidé ;
- aucun export CSV depuis le centre.

Les colonnes `pdf_report`, `html_report` et `text_report` existent dans les runs, et l'orchestrateur produit déjà des fichiers. Le problème n'est donc pas la génération initiale : **c'est l'accès utilisateur et l'organisation des rapports.**

Il faudra regrouper :

- Security Reports ;
- Executive Summaries ;
- rapports Uptime ;
- rapports Incidents ;
- rapports SLA ;
- rapports Site Intelligence.

7. Roadmap Intelligence

INTEL-003 — Reputation / blacklist

Totalement absent dans le Dashboard joint :

- Google Safe Browsing ;
- VirusTotal ;
- URLHaus ;
- PhishTank ;
- Spamhaus/DNSBL ;
- OpenPhish ;
- abuse.ch ;
- statut `CLEAN / WARNING / LISTED` ;
- tables SQL, historique et route dédiée.

C'est une fonction très visible et à forte valeur commerciale.

INTEL-004 — Search Visibility

Partiellement présent :

L'indexabilité technique peut déjà provenir du module SEO :

- `robots.txt` ;
- sitemap ;
- canonical ;
- noindex ;
- title ;
- meta description.

Il manque :

- recherche réelle sur le nom de marque ;
- pages trouvées ;
- snippets ;
- estimation des pages indexées ;
- Bing API ;
- Google Custom Search ;
- SerpAPI ;
- historique de visibilité.

INTEL-005 — News / press / brand monitoring

Totalement absent :

- RSS ;
- GDELT ;
- News API ;
- mentions de marque ;
- sentiment ;
- timeline ;
- fréquence de publication ;
- détection d'un signal de crise.

Cette fonction peut rester après la version Community initiale : elle est intéressante, mais moins vitale que Site Intelligence, les rapports et la réputation.

8. Autres fonctions du plan initial

Fonction du cahier	État
DNS / domaine / infrastructure publique	**Partiel via Site Intelligence**
SEO technique	**Partiel via Site Intelligence**
Performance web	**Partiel via Site Intelligence**
Accessibilité basique	**Absente**
Qualité éditoriale	**Absente**
Défiguration / changement	**Fondation seulement**
Réputation / blacklist	**Absente**
Search Visibility externe	**Absente hors indexabilité technique**
Presse / marque	**Absente**

Pour la défiguration, le champ `content_hash` existe déjà dans les sondes uptime. Il reste à créer le moteur de comparaison, les règles de changement légitime, les alertes et l'écran « Defacement Watch ».

9. Pages légales, footer et informations société

La page 20 du cahier n'est pas encore satisfaite.

Absent

- page ou clause About ;
- page Contact ;
- page Privacy ;
- liens dans le header ou le footer ;
- présentation Lockheed LLC et IDEO-Lab ;
- adresse légale ;
- footer complet.

Partiellement présent

Il existe déjà :

- un footer technique dans l'application ;
- un footer public sur la landing page ;
- `contact@ideo-lab.com` dans la configuration des propriétaires.

Mais le footer public affiche seulement le produit, la version et l'avertissement sur les tests autorisés. Il ne constitue pas encore le footer légal demandé.

10. Notifications et auto-test

Notifications

La cloche décrite page 21 est absente :

- pas d'icône de notification ;
- pas de compteur non lu ;
- pas de panneau des événements importants ;
- pas de liste des dernières actions terminées ;
- pas de centre des principales informations à regarder.

Les événements techniques existent déjà en SQL. Le travail principal sera donc de les sélectionner, les classer et mémoriser leur lecture.

Auto-test produit

La procédure demandée page 20 est absente.

Il faudra tester réellement :

- base centrale ;
- base privée ;
- droits d'écriture ;
- orchestrateur ;
- file des jobs ;
- outils installés ;
- génération des rapports ;
- scheduler ;
- heartbeat ;
- sondes uptime ;
- Status Page ;
- SLA ;

11. État des listes longues

Le travail est déjà commencé, mais il reste incohérent selon les pages.

Déjà équipées

- Runs : recherche, filtres, tri et pagination ;
- Findings : recherche, filtres et pagination ;
- Incidents : recherche, filtres et pagination ;
- Scheduler : recherche, filtres et pagination ;
- Jobs : filtres par année et statut.

Encore à améliorer

- Jobs : pas de vraie pagination ;
- Site Intelligence : aucun filtre, tri ou pagination ;
- Uptime : aucun filtre, tri ou pagination ;
- configuration Uptime : table potentiellement très longue ;
- Targets : aucune pagination ;
- Recent Runs du Dashboard : aucune pagination ;
- futurs Reports : tout reste à créer.

FEATURES BANKABLE

Nouveau programme

1. `RUN-KB-001E — Knowledge Base Explorer`

Création d'un véritable module visible dans le menu :

```
Security Knowledge Base
```

Avec une page dédiée comprenant :

- recherche globale ;
- filtres par sévérité, technologie, scanner, CWE, CVE et famille ;
- fiches pédagogiques complètes ;
- technologies compatibles ;
- procédures de correction ;
- commandes de validation ;
- rollback ;
- références officielles ;
- findings observés sur les véritables Runs du client.

Valeur commerciale

Le Security Analyzer ne sera plus seulement un scanner.

Il deviendra également une **encyclopédie opérationnelle de cybersécurité intégrée au contexte réel du client**.

C'est démontrable en quelques secondes et facilement présentable dans une vidéo ou une capture d'écran.

2. `RUN-REMED-002A — Remediation Command Center`

Transformation du Remediation Board actuel en véritable centre opérationnel.

Pour chaque finding :

- criticité ;
- état ;
- technologie concernée ;
- correctif recommandé ;
- propriétaire ;
- échéance ;
- difficulté estimée ;
- progression ;
- preuve avant correction ;
- preuve après correction ;
- prochain contrôle attendu.

Vue globale avec :

```
Critical blockers
Quick wins
Waiting for retest
Accepted risks
Overdue remediation
```

3. `RUN-RETEST-001A — Intelligent Retest Engine`

Aujourd'hui, un nouveau scan recrée une nouvelle photographie.

Il faut ajouter un véritable moteur de retest.

Pour chaque finding corrigé :

```
Original finding
Applied remediation
Focused retest
New evidence
Result
```

Résultats possibles :

```
CONFIRMED FIXED
STILL PRESENT
PARTIALLY FIXED
NOT TESTED
REGRESSION
```

Le système devra comparer automatiquement :

- l'ancien finding ;
- le nouveau finding ;
- les preuves ;
- la technologie ;
- la route concernée ;
- les résultats du scanner d'origine.

4. `RUN-EXEC-002A — Executive Security Report`

Création d'une page dédiée extrêmement visuelle :

```
Executive Security Report
```

Elle présentera :

- score actuel ;
- évolution du score ;
- décision globale ;
- risques bloquants ;
- risques nouveaux ;
- risques corrigés ;
- couverture réelle du scan ;
- domaines non testés ;
- remédiations en retard ;
- durée moyenne de correction ;
- prochaines actions recommandées.

Avec une lecture en trois niveaux :

```
Executive
Operational
Technical
```

5. `RUN-TREND-001A — Security Posture Timeline`

Création d'une chronologie de la sécurité du site :

```
Security posture over time
```

Avec :

- évolution du score ;
- nombre de critical/high ;
- apparition des nouvelles vulnérabilités ;
- corrections ;
- régressions ;
- changements technologiques ;
- variation de la couverture ;
- dates importantes.

La vue devra permettre de comprendre en quelques secondes :

```
Le site progresse-t-il réellement ?
```

Valeur commerciale

Une photographie unique est peu vendable.

Une évolution sur plusieurs semaines ou plusieurs mois crée une vraie logique d'abonnement.

6. `RUN-ASSET-001A — Multi-Site Security Portfolio`

Création d'une vue portefeuille pour plusieurs sites :

```
Security Portfolio
```

Chaque site affichera :

- score ;
- décision ;
- critical/high ;
- dernier scan ;
- prochaine échéance ;
- état de remédiation ;
- disponibilité ;
- SLA ;
- technologie principale ;
- tendance.

Avec classement :

```
Most exposed
Most improved
Needs immediate action
Scan overdue
Retest overdue
```

7. `RUN-EVIDENCE-001A — Evidence Vault`

Création d'un coffre de preuves par finding :

```
Detection evidence
Remediation evidence
Retest evidence
```

Avec conservation de :

- requêtes HTTP ;
- réponses ;
- extraits de logs ;
- résultats scanner ;
- date ;
- outil ;
- empreinte SHA-256 ;
- utilisateur ayant validé ;
- historique des changements.

Valeur commerciale

Cela donne au produit une dimension :

- audit ;
- conformité ;
- preuve ;
- traçabilité ;
- gouvernance.

8. `RUN-PRIORITY-001A — Business-Aware Risk Prioritization`

Aujourd'hui, une sévérité technique ne suffit pas.

Le moteur devra recalculer la priorité en intégrant :

- exposition Internet ;
- présence d'une authentification ;
- données sensibles ;
- route d'administration ;
- exploitabilité ;
- disponibilité d'un correctif ;
- technologie détectée ;
- criticité métier ;
- ancienneté ;
- récurrence ;
- existence d'une protection compensatoire.

Le résultat pourrait devenir :

```
Technical severity: MEDIUM
Operational priority: CRITICAL
Reason: public admin route + authentication exposure
```

Cyber Attack Path Engine

1. Construction automatique de la cartographie applicative

L'orchestrateur explorerait le site comme le ferait un attaquant, mais dans un cadre strictement autorisé.

Il identifierait :

- pages publiques et authentifiées ;
- formulaires et paramètres ;
- endpoints REST, GraphQL, WebSocket et AJAX ;
- fichiers JavaScript et routes découvertes dans leur contenu ;
- sous-domaines, API, CDN, stockage cloud et interfaces d'administration ;
- relations entre pages, comptes, rôles, objets métier et appels réseau ;
- identifiants prévisibles ou réutilisés ;
- chemins jamais exposés dans la navigation, mais encore accessibles.

Le résultat ne serait pas une simple liste d'URL, mais un **graphe de sécurité** :

```
DOMAIN
  ├── SUBDOMAIN
  |     ├── APPLICATION
  |     |     ├── PAGE
  |     |     ├── API ENDPOINT
  |     |     ├── AUTHENTICATION FLOW
  |     |     ├── BUSINESS OBJECT
  |     |     └── THIRD-PARTY SERVICE
  |     └── STORAGE
  └── EXTERNAL DEPENDENCY
```

2. Moteur de corrélation des vulnérabilités

Aujourd'hui, beaucoup d'outils donnent dix alertes séparées. L'orchestrateur devrait déterminer si ces dix alertes peuvent former une attaque.

Exemple :

```
Observation 1:
Une source map JavaScript est publiquement accessible.

Observation 2:
Elle révèle un ancien endpoint interne.

Observation 3:
Cet endpoint accepte un identifiant numérique.

Observation 4:
La réponse diffère selon l'existence de l'objet.

Observation 5:
Aucun contrôle d'autorisation objet n'est détecté.

Observation 6:
L'endpoint retourne des données personnelles.
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
Object enumeration
    ↓
Broken object-level authorization
    ↓
Personal data disclosure
```

Avec :

- probabilité d'exploitation ;
- impact ;
- complexité ;
- privilèges nécessaires ;
- présence ou absence d'interaction utilisateur ;
- niveau de preuve ;
- capacité de détection par les défenses existantes ;
- correctif principal ;
- contrôles compensatoires.

Ce serait beaucoup plus pertinent qu'un score CVSS isolé.

3. Analyse différentielle multi-identités

C'est probablement l'un des contrôles les plus puissants à ajouter.

L'orchestrateur utiliserait plusieurs contextes :

- visiteur anonyme ;
- utilisateur A ;
- utilisateur B ;
- administrateur ;
- compte suspendu ;
- compte partiellement validé ;
- session expirée ;
- rôle métier spécifique.

Il rejouerait les mêmes demandes et comparerait :

- codes HTTP ;
- taille et structure des réponses ;
- champs JSON ;
- objets accessibles ;
- fonctions visibles ;
- erreurs ;
- temps de réponse ;
- redirections ;
- effets en base ou dans l'interface.

Cela permettrait de détecter automatiquement :

- BOLA/IDOR ;
- élévation horizontale de privilèges ;
- élévation verticale ;
- fonctions administratives insuffisamment protégées ;
- données cachées dans les réponses ;
- contrôles réalisés seulement dans l'interface ;
- comptes désactivés conservant certains accès ;
- incohérences entre API et frontend.

Les défauts d'autorisation au niveau des objets, propriétés et fonctions figurent parmi les risques centraux du référentiel OWASP API Security. OWASP +1

4. Business Logic Abuse Analyzer

C'est là que l'outil pourrait réellement dépasser les scanners classiques.

Le moteur observerait les processus métier :

- création de compte ;
- réinitialisation du mot de passe ;
- panier et paiement ;
- coupon et promotion ;
- téléchargement ;
- vote ;
- invitation ;
- validation d'adresse électronique ;
- abonnement ;
- remboursement ;
- génération de document ;
- export de données ;
- workflow d'administration.

Il chercherait notamment :

- étapes pouvant être sautées ;
- opérations rejouables ;
- montants ou quantités modifiables ;
- coupons cumulables de façon anormale ;
- réservation sans paiement ;
- validation utilisable plusieurs fois ;
- modification d'état hors séquence ;

5. Session and Identity Forensics

Créer un sous-orchestrateur spécialisé dans les identités et sessions :

- rotation de session après connexion ;
- rotation après élévation de privilèges ;
- invalidation après déconnexion ;
- invalidation après changement de mot de passe ;
- réutilisation d'un ancien cookie ;
- coexistence de sessions ;
- durée réelle d'expiration ;
- conservation de privilèges après modification du rôle ;
- tokens présents dans URL, historique, logs ou referrer ;
- tokens JWT insuffisamment contraints ;
- incohérences entre access token et refresh token ;
- comportement du « remember me » ;
- cache navigateur sur pages sensibles ;
- fuite inter-sous-domaines liée au domaine du cookie.

L'OWASP WSTG possède des sections spécifiques sur l'authentification, les mécanismes de verrouillage, les méthodes faibles, le cache navigateur et la gestion des sessions.

OWASP +1

6. Secret and Sensitive Data Exposure Radar

Le moteur rechercherait non seulement des secrets évidents, mais aussi des **indice secrets ou de données exploitables** dans :

- HTML ;
- JavaScript ;
- source maps ;
- commentaires ;
- manifestes ;
- fichiers de configuration ;
- anciennes sauvegardes ;
- réponses d'erreur ;
- headers ;
- cookies ;
- endpoints de diagnostic ;
- documents PDF, CSV ou JSON ;
- métadonnées ;
- dépôts ou artefacts publics ;
- stockage cloud associé au domaine.

Il classifierait ce qui est découvert :

```
Credential
API key
Session material
Internal hostname
Database identifier
Personal data
Financial data
Debug information
Cloud resource
Source-code disclosure
```

La nouveauté importante serait la **validation non destructive** :

> La valeur ressemble-t-elle seulement à une clé, ou correspond-elle réellement à un service et à un périmètre encore actifs ?

Le système ne devrait jamais utiliser le secret pour réaliser des opérations invasives. Il pourrait toutefois produire un niveau de confiance basé sur son format, son emplacement, son contexte et, lorsque cela est explicitement autorisé, une vérification minimale sans modification de données.

7. Race Condition and Concurrency Analyzer

Très peu de scanners Web savent correctement tester les problèmes de concurrence.

L'orchestrateur pourrait exécuter simultanément, dans un environnement autorisé :

- deux utilisations du même coupon ;
- deux validations du même token ;
- deux changements de mot de passe ;
- deux retraits ;
- deux réservations ;
- deux demandes de remboursement ;
- deux consommations du même quota ;
- deux approbations du même objet.

Il observerait :

- doubles traitements ;
- soldes négatifs ;
- quotas dépassés ;
- objets dans un état incohérent ;
- opérations dupliquées ;
- différences entre réponse immédiate et état final.

Le test resterait désactivé par défaut ou limité à des comptes et données de test, car il peut avoir des effets métier.

8. Client-Side Trust Analyzer

Un grand nombre d'applications placent trop de confiance dans le navigateur.

L'orchestrateur analyserait les scripts pour repérer :

- contrôles effectués uniquement en JavaScript ;
- rôles et permissions contenus dans le frontend ;
- endpoints administratifs ;
- paramètres cachés ;
- prix, quotas ou statuts modifiables ;
- URLs internes ;
- clés et tokens ;
- données sensibles préchargées ;
- messages `postMessage` insuffisamment validés ;
- dépendances externes ;
- absence ou faiblesse de SRI ;
- comportement des Service Workers ;
- stockage sensible dans `localStorage`, `sessionStorage` ou IndexedDB ;
- sinks et sources DOM pouvant conduire à des XSS.

Il comparerait ensuite les règles frontend au comportement réel du backend :

```
Frontend:
max_quantity = 10

Backend:
accepts max_quantity = 100000
```

9. API Shadow and Zombie Detector

L'orchestrateur comparerait toutes les sources disponibles :

- liens HTML ;
- appels JavaScript ;
- fichiers OpenAPI ;
- schémas GraphQL ;
- historique des scans ;
- DNS ;
- réponses serveur ;
- anciennes versions d'API ;
- endpoints découverts dans les source maps ;
- fichiers robots et sitemaps ;
- journaux autorisés, lorsqu'ils sont fournis.

Il classerait :

- **documented endpoint** ;
- **undocumented endpoint** ;
- **shadow API** ;
- **deprecated but reachable** ;
- **zombie API** ;
- **duplicate API version** ;
- **orphan administrative endpoint.**

Puis il comparerait les protections entre versions :

10. Security Control Contradiction Engine

Une fonction très originale consisterait à rechercher les contradictions entre contrôles.

Exemples :

- CSP stricte, mais une page particulière est exclue ;
- authentification multifacteur affichée, mais un ancien endpoint permet de la contourner ;
- cookie `HttpOnly`, mais token dupliqué dans `localStorage` ;
- API protégée, mais export CSV public ;
- redirection HTTPS, mais contenu sensible accessible directement en HTTP sur un sous-domaine ;
- compte suspendu dans l'interface, mais token toujours accepté ;
- règle CORS restrictive sur l'API principale, mais permissive sur l'ancienne version ;
- verrouillage après cinq tentatives sur le formulaire, mais absent sur l'API ;
- WAF actif sur le domaine principal, mais origin server directement accessible.

Cette fonction produirait des constats du type :

> **Contrôle déclaré :** authentification obligatoire.
> **Contradiction observée :** endpoint historique accessible sans session.
> **Conséquence :** le contrôle existe, mais ne couvre pas toute la surface d'attaque.

11. Origin Exposure and Defense Bypass Analyzer

Le système tenterait de déterminer si les protections en façade peuvent être contournées :

- IP d'origine exposée ;
- accès direct à l'origine ;
- ancien enregistrement DNS ;
- sous-domaine non protégé ;
- certificat révélant d'autres noms ;
- CDN ou reverse proxy contournable ;
- différences de comportement entre origin et domaine public ;
- WAF présent uniquement sur certaines routes ;
- port alternatif exposé ;
- serveur de staging accessible ;
- header `Host` mal traité ;
- environnement de développement lié au même backend.

Il pourrait représenter cela ainsi :

```
INTERNET
   ↓
WAF/CDN
   ↓
PUBLIC APPLICATION
   ↓
ORIGIN SERVER

Potential bypass:
INTERNET ──────────────→ ORIGIN SERVER
```

12. Temporal Security Analyzer

Une faille n'existe pas toujours de manière permanente. L'orchestrateur devrait comparer les scans dans le temps :

- nouvelle route apparue ;
- header supprimé ;
- certificat ou configuration TLS modifié ;
- port nouvellement exposé ;
- sous-domaine créé ;
- endpoint réactivé ;
- bibliothèque devenue vulnérable ;
- réponse contenant soudainement davantage de données ;
- politique CORS élargie ;
- cookie dont les attributs ont changé ;
- réduction ou disparition de la limitation de débit.

Il générerait un **Security Drift Report** :

```
Yesterday:
Strict-Transport-Security: max-age=31536000; includeSubDomains

Today:
Header missing

Risk:
Transport security regression

Probable origin:
Deployment 2026-07-20 18:43
```

13. Evidence and Reproducibility Engine

Chaque résultat devrait contenir une preuve structurée :

- requête nettoyée ;
- réponse nettoyée ;
- horodatage ;
- identité utilisée ;
- contexte de session ;
- préconditions ;
- différence observée ;
- règle ayant déclenché le constat ;
- niveau de confiance ;
- hash des preuves ;
- procédure de reproduction ;
- proposition de correction ;
- test automatique de non-régression.

Exemple :

```
Finding:
Broken object-level authorization

Control request:
GET /api/invoices/812
Identity: USER_A
Result: 200 — USER_A owns invoice 812

Variant request:
GET /api/invoices/813
Identity: USER_A
Result: 200 — invoice belongs to USER_B
```

14. Standards Mapping Engine

Chaque contrôle de l'orchestrateur devrait être associé à :

- OWASP Top 10 2025 ;
- OWASP API Security Top 10 ;
- OWASP WSTG ;
- OWASP ASVS 5.0 ;
- CWE ;
- éventuellement CAPEC ;
- exigence interne ;
- contrôle réglementaire ou contractuel configuré par le client.

L'OWASP Top 10 actuellement publié est l'édition 2025, tandis qu'ASVS 5.0.0 a été publié en mai 2025. OWASP +2

Un finding pourrait devenir :

```
Internal:
AUTHZ-OBJECT-001

OWASP API:
API1:2023 Broken Object Level Authorization

OWASP WSTG:
Authorization Testing

OWASP ASVS:
Authorization controls

CWE:
CWE-639 Authorization Bypass Through User-Controlled Key
```

15. Remediation Verification Loop

Le cycle ne devrait pas s'arrêter au rapport.

```
DETECTED
   ↓
CONFIRMED
   ↓
ASSIGNED
   ↓
FIX PROPOSED
   ↓
RETEST SCHEDULED
   ↓
FIX VERIFIED
   ↓
NON-REGRESSION TEST CREATED
```

Après correction, l'orchestrateur rejouerait **exactement la même preuve** :

- même endpoint ;
- même utilisateur ;
- même objet ;
- même variation ;
- même règle d'évaluation.

SECURITY GUIDE BY IDEO-LAB NEW DESIGN

18 JULY 2026

AUTEUR : GUILLAUME ONEILL

v 1.8

FEATURES BANKABLE

`RUN-UX-NAV-001A — Contextual Navigation Foundation`

Premier patch sûr et immédiatement visible :

- suppression du grand `nav()` horizontal dans les pages authentifiées ;
- conservation de la sidebar comme navigation globale unique ;
- remplacement par une petite barre contextuelle propre à chaque domaine ;
- déplacement du choix de thème dans la topbar ou le menu utilisateur ;
- aucun changement de route ;
- aucune suppression de feature ;
- aucun changement SQL.

`RUN-UX-JOURNEY-001A — Guided Mission Center`

- page d'entrée orientée objectifs ;
- quatre missions principales ;
- stepper persistant ;
- prochaine action calculée depuis les données existantes ;
- mode Guided / Expert.

`RUN-UX-COMMAND-001A — Global Command Palette`

- ouverture avec `Ctrl + K` ;
- recherche des pages, cibles, scans et actions ;
- navigation rapide sans boutons supplémentaires.

`RUN-UX-NAV-001A — Contextual Navigation Foundation`

- réorganisation de la sidebar par missions ;
- suppression de l'ancien menu horizontal exhaustif ;
- nouvelle barre horizontale compacte ;
- quatre accès fixes ;
- accès populaires ;
- menu `More` ;
- conservation de toutes les routes existantes.

`RUN-UX-NAV-001B — Recent Navigation Memory`

- historique récent dans `localStorage` ;
- déduplication des destinations ;
- limite de 20 pages ;
- boutons `Back` et `History` ;
- filtrage strict des données mémorisées ;
- génération dynamique des accès récents.

`RUN-UX-NAV-001C — User Pins and SQL History`

- historique lié au compte utilisateur ;
- favoris épinglés ;
- synchronisation multi-navigateur ;
- statistiques d'utilisation ;
- règles différentes selon les rôles.

`RUN-UX-JOURNEY-001A — Guided Mission Center`

- Guided Mode ;
- Expert Mode ;
- missions Assess, Remediate, Govern et Monitor ;
- wizards réels basés sur l'état des données ;
- prochaine action recommandée.

SECURITY GUIDE BY IDEO-LAB

18 JULY 2026

AUTEUR : GUILLAUME ONEILL

v 1.8

FEATURES BANKABLE

2. `RUN-REMED-002A — Remediation Command Center`

Transformation du Remediation Board actuel en véritable centre opérationnel.

Pour chaque finding :

- criticité ;
- état ;
- technologie concernée ;
- correctif recommandé ;
- propriétaire ;
- échéance ;
- difficulté estimée ;
- progression ;
- preuve avant correction ;
- preuve après correction ;
- prochain contrôle attendu.

Vue globale avec :

```
Critical blockers
Quick wins
Waiting for retest
Accepted risks
Overdue remediation
```

3. `RUN-RETEST-001A — Intelligent Retest Engine`

Aujourd'hui, un nouveau scan recrée une nouvelle photographie.

Il faut ajouter un véritable moteur de retest.

Pour chaque finding corrigé :

```
Original finding
Applied remediation
Focused retest
New evidence
Result
```

Résultats possibles :

```
CONFIRMED FIXED
STILL PRESENT
PARTIALLY FIXED
NOT TESTED
REGRESSION
```

Le système devra comparer automatiquement :

- l'ancien finding ;
- le nouveau finding ;
- les preuves ;
- la technologie ;
- la route concernée ;
- les résultats du scanner d'origine.

4. `RUN-EXEC-002A — Executive Security Report`

Création d'une page dédiée extrêmement visuelle :

```
Executive Security Report
```

Elle présentera :

- score actuel ;
- évolution du score ;
- décision globale ;
- risques bloquants ;
- risques nouveaux ;
- risques corrigés ;
- couverture réelle du scan ;
- domaines non testés ;
- remédiations en retard ;
- durée moyenne de correction ;
- prochaines actions recommandées.

Avec une lecture en trois niveaux :

```
Executive
Operational
Technical
```

5. `RUN-TREND-001A — Security Posture Timeline`

Création d'une chronologie de la sécurité du site :

```
Security posture over time
```

Avec :

- évolution du score ;
- nombre de critical/high ;
- apparition des nouvelles vulnérabilités ;
- corrections ;
- régressions ;
- changements technologiques ;
- variation de la couverture ;
- dates importantes.

La vue devra permettre de comprendre en quelques secondes :

```
Le site progresse-t-il réellement ?
```

Valeur commerciale

Une photographie unique est peu vendable.

Une évolution sur plusieurs semaines ou plusieurs mois crée une vraie logique d'abonnement.

6. `RUN-ASSET-001A — Multi-Site Security Portfolio`

Création d'une vue portefeuille pour plusieurs sites :

```
Security Portfolio
```

Chaque site affichera :

- score ;
- décision ;
- critical/high ;
- dernier scan ;
- prochaine échéance ;
- état de remédiation ;
- disponibilité ;
- SLA ;
- technologie principale ;
- tendance.

Avec classement :

```
Most exposed
Most improved
Needs immediate action
Scan overdue
Retest overdue
```

7. `RUN-EVIDENCE-001A — Evidence Vault`

Création d'un coffre de preuves par finding :

```
Detection evidence
Remediation evidence
Retest evidence
```

Avec conservation de :

- requêtes HTTP ;
- réponses ;
- extraits de logs ;
- résultats scanner ;
- date ;
- outil ;
- empreinte SHA-256 ;
- utilisateur ayant validé ;
- historique des changements.

Valeur commerciale

Cela donne au produit une dimension :

- audit ;
- conformité ;
- preuve ;
- traçabilité ;
- gouvernance.

8. `RUN-PRIORITY-001A — Business-Aware Risk Prioritization`

Aujourd'hui, une sévérité technique ne suffit pas.

Le moteur devra recalculer la priorité en intégrant :

- exposition Internet ;
- présence d'une authentification ;
- données sensibles ;
- route d'administration ;
- exploitabilité ;
- disponibilité d'un correctif ;
- technologie détectée ;
- criticité métier ;
- ancienneté ;
- récurrence ;
- existence d'une protection compensatoire.

Le résultat pourrait devenir :

```
Technical severity: MEDIUM
Operational priority: CRITICAL
Reason: public admin route + authentication exposure
```

Cyber Attack Path Engine

1. Construction automatique de la cartographie applicative

L'orchestrateur explorerait le site comme le ferait un attaquant, mais dans un cadre strictement autorisé.

Il identifierait :

- pages publiques et authentifiées ;
- formulaires et paramètres ;
- endpoints REST, GraphQL, WebSocket et AJAX ;
- fichiers JavaScript et routes découvertes dans leur contenu ;
- sous-domaines, API, CDN, stockage cloud et interfaces d'administration ;
- relations entre pages, comptes, rôles, objets métier et appels réseau ;
- identifiants prévisibles ou réutilisés ;
- chemins jamais exposés dans la navigation, mais encore accessibles.

Le résultat ne serait pas une simple liste d'URL, mais un **graphe de sécurité** :

```
DOMAIN
  ├── SUBDOMAIN
  |     ├── APPLICATION
  |     |     ├── PAGE
  |     |     ├── API ENDPOINT
  |     |     ├── AUTHENTICATION FLOW
  |     |     ├── BUSINESS OBJECT
  |     |     └── THIRD-PARTY SERVICE
  |     └── STORAGE
  └── EXTERNAL DEPENDENCY
```

2. Moteur de corrélation des vulnérabilités

Aujourd'hui, beaucoup d'outils donnent dix alertes séparées. L'orchestrateur devrait déterminer si ces dix alertes peuvent former une attaque.

Exemple :

```
Observation 1:
Une source map JavaScript est publiquement accessible.

Observation 2:
Elle révèle un ancien endpoint interne.

Observation 3:
Cet endpoint accepte un identifiant numérique.

Observation 4:
La réponse diffère selon l'existence de l'objet.

Observation 5:
Aucun contrôle d'autorisation objet n'est détecté.

Observation 6:
L'endpoint retourne des données personnelles.
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
Object enumeration
    ↓
Broken object-level authorization
    ↓
Personal data disclosure
```

Avec :

- probabilité d'exploitation ;
- impact ;
- complexité ;
- privilèges nécessaires ;
- présence ou absence d'interaction utilisateur ;
- niveau de preuve ;
- capacité de détection par les défenses existantes ;
- correctif principal ;
- contrôles compensatoires.

Ce serait beaucoup plus pertinent qu'un score CVSS isolé.

3. Analyse différentielle multi-identités

C'est probablement l'un des contrôles les plus puissants à ajouter.

L'orchestrateur utiliserait plusieurs contextes :

- visiteur anonyme ;
- utilisateur A ;
- utilisateur B ;
- administrateur ;
- compte suspendu ;
- compte partiellement validé ;
- session expirée ;
- rôle métier spécifique.

Il rejouerait les mêmes demandes et comparerait :

- codes HTTP ;
- taille et structure des réponses ;
- champs JSON ;
- objets accessibles ;
- fonctions visibles ;
- erreurs ;
- temps de réponse ;
- redirections ;
- effets en base ou dans l'interface.

Cela permettrait de détecter automatiquement :

- BOLA/IDOR ;
- élévation horizontale de privilèges ;
- élévation verticale ;
- fonctions administratives insuffisamment protégées ;
- données cachées dans les réponses ;
- contrôles réalisés seulement dans l'interface ;
- comptes désactivés conservant certains accès ;
- incohérences entre API et frontend.

Les défauts d'autorisation au niveau des objets, propriétés et fonctions figurent parmi les risques centraux du référentiel OWASP API Security. OWASP +1

4. Business Logic Abuse Analyzer

C'est là que l'outil pourrait réellement dépasser les scanners classiques.

Le moteur observerait les processus métier :

- création de compte ;
- réinitialisation du mot de passe ;
- panier et paiement ;
- coupon et promotion ;
- téléchargement ;
- vote ;
- invitation ;
- validation d'adresse électronique ;
- abonnement ;
- remboursement ;
- génération de document ;
- export de données ;
- workflow d'administration.

Il chercherait notamment :

- étapes pouvant être sautées ;
- opérations rejouables ;
- montants ou quantités modifiables ;
- coupons cumulables de façon anormale ;
- réservation sans paiement ;
- validation utilisable plusieurs fois ;
- modification d'état hors séquence ;

5. Session and Identity Forensics

Créer un sous-orchestrateur spécialisé dans les identités et sessions :

- rotation de session après connexion ;
- rotation après élévation de privilèges ;
- invalidation après déconnexion ;
- invalidation après changement de mot de passe ;
- réutilisation d'un ancien cookie ;
- coexistence de sessions ;
- durée réelle d'expiration ;
- conservation de privilèges après modification du rôle ;
- tokens présents dans URL, historique, logs ou referrer ;
- tokens JWT insuffisamment contraints ;
- incohérences entre access token et refresh token ;
- comportement du « remember me » ;
- cache navigateur sur pages sensibles ;
- fuite inter-sous-domaines liée au domaine du cookie.

L'OWASP WSTG possède des sections spécifiques sur l'authentification, les mécanismes de verrouillage, les méthodes faibles, le cache navigateur et la gestion des sessions.

OWASP +1

6. Secret and Sensitive Data Exposure Radar

Le moteur rechercherait non seulement des secrets évidents, mais aussi des **indice secrets ou de données exploitables** dans :

- HTML ;
- JavaScript ;
- source maps ;
- commentaires ;
- manifestes ;
- fichiers de configuration ;
- anciennes sauvegardes ;
- réponses d'erreur ;
- headers ;
- cookies ;
- endpoints de diagnostic ;
- documents PDF, CSV ou JSON ;
- métadonnées ;
- dépôts ou artefacts publics ;
- stockage cloud associé au domaine.

Il classifierait ce qui est découvert :

```
Credential
API key
Session material
Internal hostname
Database identifier
Personal data
Financial data
Debug information
Cloud resource
Source-code disclosure
```

La nouveauté importante serait la **validation non destructive** :

> La valeur ressemble-t-elle seulement à une clé, ou correspond-elle réellement à un service et à un périmètre encore actifs ?

Le système ne devrait jamais utiliser le secret pour réaliser des opérations invasives. Il pourrait toutefois produire un niveau de confiance basé sur son format, son emplacement, son contexte et, lorsque cela est explicitement autorisé, une vérification minimale sans modification de données.

7. Race Condition and Concurrency Analyzer

Très peu de scanners Web savent correctement tester les problèmes de concurrence.

L'orchestrateur pourrait exécuter simultanément, dans un environnement autorisé :

- deux utilisations du même coupon ;
- deux validations du même token ;
- deux changements de mot de passe ;
- deux retraits ;
- deux réservations ;
- deux demandes de remboursement ;
- deux consommations du même quota ;
- deux approbations du même objet.

Il observerait :

- doubles traitements ;
- soldes négatifs ;
- quotas dépassés ;
- objets dans un état incohérent ;
- opérations dupliquées ;
- différences entre réponse immédiate et état final.

Le test resterait désactivé par défaut ou limité à des comptes et données de test, car il peut avoir des effets métier.

8. Client-Side Trust Analyzer

Un grand nombre d'applications placent trop de confiance dans le navigateur.

L'orchestrateur analyserait les scripts pour repérer :

- contrôles effectués uniquement en JavaScript ;
- rôles et permissions contenus dans le frontend ;
- endpoints administratifs ;
- paramètres cachés ;
- prix, quotas ou statuts modifiables ;
- URLs internes ;
- clés et tokens ;
- données sensibles préchargées ;
- messages `postMessage` insuffisamment validés ;
- dépendances externes ;
- absence ou faiblesse de SRI ;
- comportement des Service Workers ;
- stockage sensible dans `localStorage`, `sessionStorage` ou IndexedDB ;
- sinks et sources DOM pouvant conduire à des XSS.

Il comparerait ensuite les règles frontend au comportement réel du backend :

```
Frontend:
max_quantity = 10

Backend:
accepts max_quantity = 100000
```

9. API Shadow and Zombie Detector

L'orchestrateur comparerait toutes les sources disponibles :

- liens HTML ;
- appels JavaScript ;
- fichiers OpenAPI ;
- schémas GraphQL ;
- historique des scans ;
- DNS ;
- réponses serveur ;
- anciennes versions d'API ;
- endpoints découverts dans les source maps ;
- fichiers robots et sitemaps ;
- journaux autorisés, lorsqu'ils sont fournis.

Il classerait :

- **documented endpoint** ;
- **undocumented endpoint** ;
- **shadow API** ;
- **deprecated but reachable** ;
- **zombie API** ;
- **duplicate API version** ;
- **orphan administrative endpoint**.

Puis il comparerait les protections entre versions :

10. Security Control Contradiction Engine

Une fonction très originale consisterait à rechercher les contradictions entre contrôles.

Exemples :

- CSP stricte, mais une page particulière est exclue ;
- authentification multifacteur affichée, mais un ancien endpoint permet de la contourner ;
- cookie `HttpOnly`, mais token dupliqué dans `localStorage` ;
- API protégée, mais export CSV public ;
- redirection HTTPS, mais contenu sensible accessible directement en HTTP sur un sous-domaine ;
- compte suspendu dans l'interface, mais token toujours accepté ;
- règle CORS restrictive sur l'API principale, mais permissive sur l'ancienne version ;
- verrouillage après cinq tentatives sur le formulaire, mais absent sur l'API ;
- WAF actif sur le domaine principal, mais origin server directement accessible.

Cette fonction produirait des constats du type :

> **Contrôle déclaré :** authentification obligatoire.
> **Contradiction observée :** endpoint historique accessible sans session.
> **Conséquence :** le contrôle existe, mais ne couvre pas toute la surface d'attaque.

11. Origin Exposure and Defense Bypass Analyzer

Le système tenterait de déterminer si les protections en façade peuvent être contournées :

- IP d'origine exposée ;
- accès direct à l'origine ;
- ancien enregistrement DNS ;
- sous-domaine non protégé ;
- certificat révélant d'autres noms ;
- CDN ou reverse proxy contournable ;
- différences de comportement entre origin et domaine public ;
- WAF présent uniquement sur certaines routes ;
- port alternatif exposé ;
- serveur de staging accessible ;
- header `Host` mal traité ;
- environnement de développement lié au même backend.

Il pourrait représenter cela ainsi :

```
INTERNET
   ↓
WAF/CDN
   ↓
PUBLIC APPLICATION
   ↓
ORIGIN SERVER

Potential bypass:
INTERNET ──────────────→ ORIGIN SERVER
```

12. Temporal Security Analyzer

Une faille n'existe pas toujours de manière permanente. L'orchestrateur devrait comparer les scans dans le temps :

- nouvelle route apparue ;
- header supprimé ;
- certificat ou configuration TLS modifié ;
- port nouvellement exposé ;
- sous-domaine créé ;
- endpoint réactivé ;
- bibliothèque devenue vulnérable ;
- réponse contenant soudainement davantage de données ;
- politique CORS élargie ;
- cookie dont les attributs ont changé ;
- réduction ou disparition de la limitation de débit.

Il générerait un **Security Drift Report** :

```
Yesterday:
Strict-Transport-Security: max-age=31536000; includeSubDomains

Today:
Header missing

Risk:
Transport security regression

Probable origin:
Deployment 2026-07-20 18:43
```

13. Evidence and Reproducibility Engine

Chaque résultat devrait contenir une preuve structurée :

- requête nettoyée ;
- réponse nettoyée ;
- horodatage ;
- identité utilisée ;
- contexte de session ;
- préconditions ;
- différence observée ;
- règle ayant déclenché le constat ;
- niveau de confiance ;
- hash des preuves ;
- procédure de reproduction ;
- proposition de correction ;
- test automatique de non-régression.

Exemple :

```
Finding:
Broken object-level authorization

Control request:
GET /api/invoices/812
Identity: USER_A
Result: 200 — USER_A owns invoice 812

Variant request:
GET /api/invoices/813
Identity: USER_A
Result: 200 — invoice belongs to USER_B
```

14. Standards Mapping Engine

Chaque contrôle de l'orchestrateur devrait être associé à :

- OWASP Top 10 2025 ;
- OWASP API Security Top 10 ;
- OWASP WSTG ;
- OWASP ASVS 5.0 ;
- CWE ;
- éventuellement CAPEC ;
- exigence interne ;
- contrôle réglementaire ou contractuel configuré par le client.

L'OWASP Top 10 actuellement publié est l'édition 2025, tandis qu'ASVS 5.0.0 a été publié en mai 2025. OWASP +2

Un finding pourrait devenir :

```
Internal:
AUTHZ-OBJECT-001

OWASP API:
API1:2023 Broken Object Level Authorization

OWASP WSTG:
Authorization Testing

OWASP ASVS:
Authorization controls

CWE:
CWE-639 Authorization Bypass Through User-Controlled Key
```

15. Remediation Verification Loop

Le cycle ne devrait pas s'arrêter au rapport.

```
DETECTED
   ↓
CONFIRMED
   ↓
ASSIGNED
   ↓
FIX PROPOSED
   ↓
RETEST SCHEDULED
   ↓
FIX VERIFIED
   ↓
NON-REGRESSION TEST CREATED
```

Après correction, l'orchestrateur rejouerait **exactement la même preuve** :

- même endpoint ;
- même utilisateur ;
- même objet ;
- même variation ;
- même règle d'évaluation.

1. Créer un fichier protégé contenant la configuration SMTP

Sur le serveur Linux :

Bash

```bash
sudo mkdir -p /etc/ideo-security
sudo nano /etc/ideo-security/signup-mail.env
```

Insérer exactement :

INI

```ini
IDEO_SECURITY_PUBLIC_URL=https://analyzer.ideo-lab.com

IDEO_SECURITY_SIGNUP_FROM_EMAIL=contact@ideo-lab.com
IDEO_SECURITY_SIGNUP_FROM_NAME="IDEO-Lab Security Analyzer"

IDEO_SECURITY_SIGNUP_SMTP_HOST=smtp.ions.fr
IDEO_SECURITY_SIGNUP_SMTP_PORT=587
IDEO_SECURITY_SIGNUP_SMTP_USER=contact@ideo-lab.com
IDEO_SECURITY_SIGNUP_SMTP_PASSWORD="TON_NOUVEAU_MOT_DE_PASSE"

IDEO_SECURITY_SIGNUP_SMTP_SSL=false
IDEO_SECURITY_SIGNUP_SMTP_STARTTLS=true
```

Remplacer uniquement :

SECURITY GUIDE BY IDEO-LAB

18 JULY 2026

AUTEUR : GUILLAUME ONEILL

v 1.8

FEATURES BANKABLE

2. `RUN-REMED-002A — Remediation Command Center`

Transformation du Remediation Board actuel en véritable centre opérationnel.

Pour chaque finding :

- criticité ;
- état ;
- technologie concernée ;
- correctif recommandé ;
- propriétaire ;
- échéance ;
- difficulté estimée ;
- progression ;
- preuve avant correction ;
- preuve après correction ;
- prochain contrôle attendu.

Vue globale avec :

```
Critical blockers
Quick wins
Waiting for retest
Accepted risks
Overdue remediation
```

3. `RUN-RETEST-001A — Intelligent Retest Engine`

Aujourd'hui, un nouveau scan recrée une nouvelle photographie.

Il faut ajouter un véritable moteur de retest.

Pour chaque finding corrigé :

```
Original finding
Applied remediation
Focused retest
New evidence
Result
```

Résultats possibles :

```
CONFIRMED FIXED
STILL PRESENT
PARTIALLY FIXED
NOT TESTED
REGRESSION
```

Le système devra comparer automatiquement :

- l'ancien finding ;
- le nouveau finding ;
- les preuves ;
- la technologie ;
- la route concernée ;
- les résultats du scanner d'origine.

4. `RUN-EXEC-002A — Executive Security Report`

Création d'une page dédiée extrêmement visuelle :

```
Executive Security Report
```

Elle présentera :

- score actuel ;
- évolution du score ;
- décision globale ;
- risques bloquants ;
- risques nouveaux ;
- risques corrigés ;
- couverture réelle du scan ;
- domaines non testés ;
- remédiations en retard ;
- durée moyenne de correction ;
- prochaines actions recommandées.

Avec une lecture en trois niveaux :

```
Executive
Operational
Technical
```

5. `RUN-TREND-001A — Security Posture Timeline`

Création d'une chronologie de la sécurité du site :

```
Security posture over time
```

Avec :

- évolution du score ;
- nombre de critical/high ;
- apparition des nouvelles vulnérabilités ;
- corrections ;
- régressions ;
- changements technologiques ;
- variation de la couverture ;
- dates importantes.

La vue devra permettre de comprendre en quelques secondes :

```
Le site progresse-t-il réellement ?
```

Valeur commerciale

Une photographie unique est peu vendable.

Une évolution sur plusieurs semaines ou plusieurs mois crée une vraie logique d'abonnement.

6. `RUN-ASSET-001A — Multi-Site Security Portfolio`

Création d'une vue portefeuille pour plusieurs sites :

```
Security Portfolio
```

Chaque site affichera :

- score ;
- décision ;
- critical/high ;
- dernier scan ;
- prochaine échéance ;
- état de remédiation ;
- disponibilité ;
- SLA ;
- technologie principale ;
- tendance.

Avec classement :

```
Most exposed
Most improved
Needs immediate action
Scan overdue
Retest overdue
```

7. `RUN-EVIDENCE-001A — Evidence Vault`

Création d'un coffre de preuves par finding :

```
Detection evidence
Remediation evidence
Retest evidence
```

Avec conservation de :

- requêtes HTTP ;
- réponses ;
- extraits de logs ;
- résultats scanner ;
- date ;
- outil ;
- empreinte SHA-256 ;
- utilisateur ayant validé ;
- historique des changements.

Valeur commerciale

Cela donne au produit une dimension :

- audit ;
- conformité ;
- preuve ;
- traçabilité ;
- gouvernance.

8. `RUN-PRIORITY-001A — Business-Aware Risk Prioritization`

Aujourd'hui, une sévérité technique ne suffit pas.

Le moteur devra recalculer la priorité en intégrant :

- exposition Internet ;
- présence d'une authentification ;
- données sensibles ;
- route d'administration ;
- exploitabilité ;
- disponibilité d'un correctif ;
- technologie détectée ;
- criticité métier ;
- ancienneté ;
- récurrence ;
- existence d'une protection compensatoire.

Le résultat pourrait devenir :

```
Technical severity: MEDIUM
Operational priority: CRITICAL
Reason: public admin route + authentication exposure
```

Cyber Attack Path Engine

1. Construction automatique de la cartographie applicative

L'orchestrateur explorerait le site comme le ferait un attaquant, mais dans un cadre strictement autorisé.

Il identifierait :

- pages publiques et authentifiées ;
- formulaires et paramètres ;
- endpoints REST, GraphQL, WebSocket et AJAX ;
- fichiers JavaScript et routes découvertes dans leur contenu ;
- sous-domaines, API, CDN, stockage cloud et interfaces d'administration ;
- relations entre pages, comptes, rôles, objets métier et appels réseau ;
- identifiants prévisibles ou réutilisés ;
- chemins jamais exposés dans la navigation, mais encore accessibles.

Le résultat ne serait pas une simple liste d'URL, mais un **graphe de sécurité** :

```
DOMAIN
  ├── SUBDOMAIN
  |     ├── APPLICATION
  |     |     ├── PAGE
  |     |     ├── API ENDPOINT
  |     |     ├── AUTHENTICATION FLOW
  |     |     ├── BUSINESS OBJECT
  |     |     └── THIRD-PARTY SERVICE
  |     └── STORAGE
  └── EXTERNAL DEPENDENCY
```

2. Moteur de corrélation des vulnérabilités

Aujourd'hui, beaucoup d'outils donnent dix alertes séparées. L'orchestrateur devrait déterminer si ces dix alertes peuvent former une attaque.

Exemple :

```
Observation 1:
Une source map JavaScript est publiquement accessible.

Observation 2:
Elle révèle un ancien endpoint interne.

Observation 3:
Cet endpoint accepte un identifiant numérique.

Observation 4:
La réponse diffère selon l'existence de l'objet.

Observation 5:
Aucun contrôle d'autorisation objet n'est détecté.

Observation 6:
L'endpoint retourne des données personnelles.
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
```

Le moteur créerait alors :

```
ATTACK PATH AP-0042

Public JavaScript
    ↓
Endpoint discovery
    ↓
Object enumeration
    ↓
Broken object-level authorization
    ↓
Personal data disclosure
```

Avec :

- probabilité d'exploitation ;
- impact ;
- complexité ;
- privilèges nécessaires ;
- présence ou absence d'interaction utilisateur ;
- niveau de preuve ;
- capacité de détection par les défenses existantes ;
- correctif principal ;
- contrôles compensatoires.

Ce serait beaucoup plus pertinent qu'un score CVSS isolé.

3. Analyse différentielle multi-identités

C'est probablement l'un des contrôles les plus puissants à ajouter.

L'orchestrateur utiliserait plusieurs contextes :

- visiteur anonyme ;
- utilisateur A ;
- utilisateur B ;
- administrateur ;
- compte suspendu ;
- compte partiellement validé ;
- session expirée ;
- rôle métier spécifique.

Il rejouerait les mêmes demandes et comparerait :

- codes HTTP ;
- taille et structure des réponses ;
- champs JSON ;
- objets accessibles ;
- fonctions visibles ;
- erreurs ;
- temps de réponse ;
- redirections ;
- effets en base ou dans l'interface.

Cela permettrait de détecter automatiquement :

- BOLA/IDOR ;
- élévation horizontale de privilèges ;
- élévation verticale ;
- fonctions administratives insuffisamment protégées ;
- données cachées dans les réponses ;
- contrôles réalisés seulement dans l'interface ;
- comptes désactivés conservant certains accès ;
- incohérences entre API et frontend.

Les défauts d'autorisation au niveau des objets, propriétés et fonctions figurent parmi les risques centraux du référentiel OWASP API Security. OWASP +1

4. Business Logic Abuse Analyzer

C'est là que l'outil pourrait réellement dépasser les scanners classiques.

Le moteur observerait les processus métier :

- création de compte ;
- réinitialisation du mot de passe ;
- panier et paiement ;
- coupon et promotion ;
- téléchargement ;
- vote ;
- invitation ;
- validation d'adresse électronique ;
- abonnement ;
- remboursement ;
- génération de document ;
- export de données ;
- workflow d'administration.

Il chercherait notamment :

- étapes pouvant être sautées ;
- opérations rejouables ;
- montants ou quantités modifiables ;
- coupons cumulables de façon anormale ;
- réservation sans paiement ;
- validation utilisable plusieurs fois ;
- modification d'état hors séquence ;

5. Session and Identity Forensics

Créer un sous-orchestrateur spécialisé dans les identités et sessions :

- rotation de session après connexion ;
- rotation après élévation de privilèges ;
- invalidation après déconnexion ;
- invalidation après changement de mot de passe ;
- réutilisation d'un ancien cookie ;
- coexistence de sessions ;
- durée réelle d'expiration ;
- conservation de privilèges après modification du rôle ;
- tokens présents dans URL, historique, logs ou referrer ;
- tokens JWT insuffisamment contraints ;
- incohérences entre access token et refresh token ;
- comportement du « remember me » ;
- cache navigateur sur pages sensibles ;
- fuite inter-sous-domaines liée au domaine du cookie.

L'OWASP WSTG possède des sections spécifiques sur l'authentification, les mécanismes de verrouillage, les méthodes faibles, le cache navigateur et la gestion des sessions.

OWASP +1

6. Secret and Sensitive Data Exposure Radar

Le moteur rechercherait non seulement des secrets évidents, mais aussi des **indice secrets ou de données exploitables** dans :

- HTML ;
- JavaScript ;
- source maps ;
- commentaires ;
- manifestes ;
- fichiers de configuration ;
- anciennes sauvegardes ;
- réponses d'erreur ;
- headers ;
- cookies ;
- endpoints de diagnostic ;
- documents PDF, CSV ou JSON ;
- métadonnées ;
- dépôts ou artefacts publics ;
- stockage cloud associé au domaine.

Il classifierait ce qui est découvert :

```
Credential
API key
Session material
Internal hostname
Database identifier
Personal data
Financial data
Debug information
Cloud resource
Source-code disclosure
```

La nouveauté importante serait la **validation non destructive** :

> La valeur ressemble-t-elle seulement à une clé, ou correspond-elle réellement à un service et à un périmètre encore actifs ?

Le système ne devrait jamais utiliser le secret pour réaliser des opérations invasives. Il pourrait toutefois produire un niveau de confiance basé sur son format, son emplacement, son contexte et, lorsque cela est explicitement autorisé, une vérification minimale sans modification de données.

7. Race Condition and Concurrency Analyzer

Très peu de scanners Web savent correctement tester les problèmes de concurrence.

L'orchestrateur pourrait exécuter simultanément, dans un environnement autorisé :

- deux utilisations du même coupon ;
- deux validations du même token ;
- deux changements de mot de passe ;
- deux retraits ;
- deux réservations ;
- deux demandes de remboursement ;
- deux consommations du même quota ;
- deux approbations du même objet.

Il observerait :

- doubles traitements ;
- soldes négatifs ;
- quotas dépassés ;
- objets dans un état incohérent ;
- opérations dupliquées ;
- différences entre réponse immédiate et état final.

Le test resterait désactivé par défaut ou limité à des comptes et données de test, car il peut avoir des effets métier.

8. Client-Side Trust Analyzer

Un grand nombre d'applications placent trop de confiance dans le navigateur.

L'orchestrateur analyserait les scripts pour repérer :

- contrôles effectués uniquement en JavaScript ;
- rôles et permissions contenus dans le frontend ;
- endpoints administratifs ;
- paramètres cachés ;
- prix, quotas ou statuts modifiables ;
- URLs internes ;
- clés et tokens ;
- données sensibles préchargées ;
- messages `postMessage` insuffisamment validés ;
- dépendances externes ;
- absence ou faiblesse de SRI ;
- comportement des Service Workers ;
- stockage sensible dans `localStorage`, `sessionStorage` ou IndexedDB ;
- sinks et sources DOM pouvant conduire à des XSS.

Il comparerait ensuite les règles frontend au comportement réel du backend :

```
Frontend:
max_quantity = 10

Backend:
accepts max_quantity = 100000
```

9. API Shadow and Zombie Detector

L'orchestrateur comparerait toutes les sources disponibles :

- liens HTML ;
- appels JavaScript ;
- fichiers OpenAPI ;
- schémas GraphQL ;
- historique des scans ;
- DNS ;
- réponses serveur ;
- anciennes versions d'API ;
- endpoints découverts dans les source maps ;
- fichiers robots et sitemaps ;
- journaux autorisés, lorsqu'ils sont fournis.

Il classerait :

- **documented endpoint** ;
- **undocumented endpoint** ;
- **shadow API** ;
- **deprecated but reachable** ;
- **zombie API** ;
- **duplicate API version** ;
- **orphan administrative endpoint**.

Puis il comparerait les protections entre versions :

10. Security Control Contradiction Engine

Une fonction très originale consisterait à rechercher les contradictions entre contrôles.

Exemples :

- CSP stricte, mais une page particulière est exclue ;
- authentification multifacteur affichée, mais un ancien endpoint permet de la contourner ;
- cookie `HttpOnly`, mais token dupliqué dans `localStorage` ;
- API protégée, mais export CSV public ;
- redirection HTTPS, mais contenu sensible accessible directement en HTTP sur un sous-domaine ;
- compte suspendu dans l'interface, mais token toujours accepté ;
- règle CORS restrictive sur l'API principale, mais permissive sur l'ancienne version ;
- verrouillage après cinq tentatives sur le formulaire, mais absent sur l'API ;
- WAF actif sur le domaine principal, mais origin server directement accessible.

Cette fonction produirait des constats du type :

> **Contrôle déclaré :** authentification obligatoire.
> **Contradiction observée :** endpoint historique accessible sans session.
> **Conséquence :** le contrôle existe, mais ne couvre pas toute la surface d'attaque.

11. Origin Exposure and Defense Bypass Analyzer

Le système tenterait de déterminer si les protections en façade peuvent être contournées :

- IP d'origine exposée ;
- accès direct à l'origine ;
- ancien enregistrement DNS ;
- sous-domaine non protégé ;
- certificat révélant d'autres noms ;
- CDN ou reverse proxy contournable ;
- différences de comportement entre origin et domaine public ;
- WAF présent uniquement sur certaines routes ;
- port alternatif exposé ;
- serveur de staging accessible ;
- header `Host` mal traité ;
- environnement de développement lié au même backend.

Il pourrait représenter cela ainsi :

```
INTERNET
   ↓
WAF/CDN
   ↓
PUBLIC APPLICATION
   ↓
ORIGIN SERVER

Potential bypass:
INTERNET ──────────────→ ORIGIN SERVER
```

12. Temporal Security Analyzer

Une faille n'existe pas toujours de manière permanente. L'orchestrateur devrait comparer les scans dans le temps :

- nouvelle route apparue ;
- header supprimé ;
- certificat ou configuration TLS modifié ;
- port nouvellement exposé ;
- sous-domaine créé ;
- endpoint réactivé ;
- bibliothèque devenue vulnérable ;
- réponse contenant soudainement davantage de données ;
- politique CORS élargie ;
- cookie dont les attributs ont changé ;
- réduction ou disparition de la limitation de débit.

Il générerait un **Security Drift Report** :

```
Yesterday:
Strict-Transport-Security: max-age=31536000; includeSubDomains

Today:
Header missing

Risk:
Transport security regression

Probable origin:
Deployment 2026-07-20 18:43
```

13. Evidence and Reproducibility Engine

Chaque résultat devrait contenir une preuve structurée :

- requête nettoyée ;
- réponse nettoyée ;
- horodatage ;
- identité utilisée ;
- contexte de session ;
- préconditions ;
- différence observée ;
- règle ayant déclenché le constat ;
- niveau de confiance ;
- hash des preuves ;
- procédure de reproduction ;
- proposition de correction ;
- test automatique de non-régression.

Exemple :

```
Finding:
Broken object-level authorization

Control request:
GET /api/invoices/812
Identity: USER_A
Result: 200 — USER_A owns invoice 812

Variant request:
GET /api/invoices/813
Identity: USER_A
Result: 200 — invoice belongs to USER_B
```

14. Standards Mapping Engine

Chaque contrôle de l'orchestrateur devrait être associé à :

- OWASP Top 10 2025 ;
- OWASP API Security Top 10 ;
- OWASP WSTG ;
- OWASP ASVS 5.0 ;
- CWE ;
- éventuellement CAPEC ;
- exigence interne ;
- contrôle réglementaire ou contractuel configuré par le client.

L'OWASP Top 10 actuellement publié est l'édition 2025, tandis qu'ASVS 5.0.0 a été publié en mai 2025. OWASP +2

Un finding pourrait devenir :

```
Internal:
AUTHZ-OBJECT-001

OWASP API:
API1:2023 Broken Object Level Authorization

OWASP WSTG:
Authorization Testing

OWASP ASVS:
Authorization controls

CWE:
CWE-639 Authorization Bypass Through User-Controlled Key
```

15. Remediation Verification Loop

Le cycle ne devrait pas s'arrêter au rapport.

```
DETECTED
   ↓
CONFIRMED
   ↓
ASSIGNED
   ↓
FIX PROPOSED
   ↓
RETEST SCHEDULED
   ↓
FIX VERIFIED
   ↓
NON-REGRESSION TEST CREATED
```

Après correction, l'orchestrateur rejouerait **exactement la même preuve** :

- même endpoint ;
- même utilisateur ;
- même objet ;
- même variation ;
- même règle d'évaluation.

1. Créer un fichier protégé contenant la configuration SMTP

Sur le serveur Linux :

Bash

```bash
sudo mkdir -p /etc/ideo-security
sudo nano /etc/ideo-security/signup-mail.env
```

Insérer exactement :

INI

```ini
IDEO_SECURITY_PUBLIC_URL=https://analyzer.ideo-lab.com

IDEO_SECURITY_SIGNUP_FROM_EMAIL=contact@ideo-lab.com
IDEO_SECURITY_SIGNUP_FROM_NAME="IDEO-Lab Security Analyzer"

IDEO_SECURITY_SIGNUP_SMTP_HOST=smtp.ions.fr
IDEO_SECURITY_SIGNUP_SMTP_PORT=587
IDEO_SECURITY_SIGNUP_SMTP_USER=contact@ideo-lab.com
IDEO_SECURITY_SIGNUP_SMTP_PASSWORD="TON_NOUVEAU_MOT_DE_PASSE"

IDEO_SECURITY_SIGNUP_SMTP_SSL=false
IDEO_SECURITY_SIGNUP_SMTP_STARTTLS=true
```

Remplacer uniquement :

MOCKUP

Security Analyzer Dashboard

Your security and availability cockpit.

Latest scanned target: https://www.everestrank.com

GENERAL | SETUP | WORKFLOW

LAST SCAN | LAST RUNS | RECENT RUNS

+ LAUNCH NEW SCAN

Start a security assessment

GUIDED WORKFLOW

Progress through the product logically

RECOMMENDED NOW
Open remediation

1 ASSESSMENT
Assess
Launch an authorized scan and follow its execution until the result is stored.

Start assessment

Scan jobs Runs

2 EVIDENCE
Understand
Interpret posture, findings, attack surface and operational risk.

Open summary

Findings Attack surface
Risk matrix

3 ACTION RECOMMENDED
Remediate
Assign corrective work, document decisions and prepare validation.

Open remediation

Knowledge base

4 VERIFICATION
Validate
Retest corrected findings and measure progress between comparable assessments.

Compare runs

Posture timeline Launch retest

5 MANAGEMENT
Govern
Consolidate target posture and prepare compliance and executive decisions.

Open portfolio

Compliance Executive report

CONTINUOUS OPERATIONS

Monitor availability
Uptime, incidents, SLA and the public status page continue in parallel with the security workflow.

Availability | Incidents | SLA | Status page | Configure

PUBLIC CONTEXT

Site Intelligence
Review DNS, HTTP, SEO, indexability and public performance information.

Open Site Intelligence

AMELIORER LE DESIGN

1. Créer un véritable « mode de parcours » dynamique

La homepage ne devrait pas être exactement la même selon la situation du compte.

Compte neuf

Le message principal devient :

Welcome to Security Analyzer

`Step 1 of 5 — Create your first security baseline`

Bouton principal :

Launch your first scan

Actions secondaires discrètes :

`Configure uptime` · `Learn how scans work`

Le workflow affiche alors clairement :

```
[1 Assess — CURRENT] → [2 Understand] → [3 Remediate] → [4 Validate] → [5 Govern]
```

Scan en cours

Le bouton `Launch new scan` ne doit plus être l'action dominante. Il devient :

Scan in progress — 63%

Avec :

- la cible ;
- le profil ;
- l'étape actuelle ;
- un bouton `Follow live scan`.

Scan terminé avec des risques critiques

La homepage devient orientée décision :

Your assessment found 2 blocking security issues

Bouton principal :

Review critical findings

Boutons secondaires :

`Open summary` · `View complete scan`

Le workflow met en évidence `Understand` ou `Remediate`, selon l'état réel des findings.

Corrections prêtes à être validées

Le message devient :

Corrections are ready for validation

Bouton principal :

Launch validation scan

Le workflow met alors `Validate` en évidence.

Le code possède déjà les fondations nécessaires : calcul d'une prochaine action, onboarding, checklist de première utilisation et workflow recommandé. Il faut maintenant les fusionner dans une expérience unique plutôt que les afficher comme plusieurs guides concurrents. security_dashboard_app

2. Transformer le workflow en indicateur « Vous êtes ici »

Le workflow actuel est beau, mais il ressemble encore à cinq menus placés côte à côte.

Je proposerais d'ajouter au-dessus des cartes une ligne très explicite :

```
CURRENT PHASE
3 of 5 — Remediate priority findings
```

La carte active aurait :

- `CURRENT STEP` au lieu de seulement `RECOMMENDED` ;
- une couleur cyan plus forte ;
- les étapes terminées avec une coche ;
- les étapes futures légèrement atténuées.

Exemple :

```
✓ Assess      ✓ Understand      3 Remediate      4 Validate      5 Govern
Completed       Completed         Current          Next           Later
```

Cela permet à l'utilisateur de comprendre immédiatement :

- ce qu'il a déjà fait ;
- où il se trouve ;
- ce qu'il doit faire maintenant ;
- ce qui arrivera ensuite.

3. Fusionner « Start here » et « Next best action »

Actuellement, le bloc supérieur dit `Start here`, alors que plus bas une autre carte dit `Next best action`.

Ces deux zones remplissent pratiquement la même fonction.

Je recommande :

- de conserver la zone de droite du bandeau ;
- mais de la rendre **dynamique** ;
- de supprimer ensuite la grande carte `Next best action` du bloc inférieur, ou de la transformer en résumé très compact.

Exemple dans ta situation actuelle :

```
RECOMMENDED NOW

Fix 2 blocking findings
Your latest assessment is NO-GO with
2 critical and 3 high findings.

[ Open remediation ]
View executive summary
```

Le bouton `Launch new scan` pourrait rester disponible comme action secondaire, mais il ne devrait pas être visuellement prioritaire lorsque le système sait déjà que la prochaine action logique est de corriger les findings.

4. Donner un rôle plus clair aux six onglets

Les onglets sont maintenant bien placés, mais leurs noms mélangent plusieurs concepts :

```
GENERAL
SETUP
WORKFLOW
LAST SCAN
LAST RUNS
RECENT RUNS
```

Je verrais une terminologie plus immédiatement compréhensible :

```
OVERVIEW        GET STARTED
WORKFLOW        LATEST SCAN
SCAN HISTORY    RECENT ACTIVITY
```

`Last Runs` et `Recent Runs` sont trop proches sémantiquement pour un nouvel utilisateur. Il ne saura probablement pas faire la différence.

Il faudrait choisir entre :

- `Scan history` : l'ensemble des scans ;
- `Recent activity` : scans, jobs, incidents, changements de findings et actions utilisateur.

Cela rendrait les six onglets réellement complémentaires.

5. Rendre les indicateurs inférieurs entièrement actionnables

Les cartes :

- Security posture ;
- Blocking findings ;
- Availability ;
- Incidents ;

doivent devenir des **portes d'entrée complètes**, pas seulement des zones contenant un petit lien.

Je recommande que toute la carte soit cliquable, avec :

- un léger effet de survol ;
- une flèche discrète à droite ;
- une explication du chiffre ;
- une action correspondant à l'état.

Par exemple :

```
SECURITY POSTURE                    →
28 / 100
NO-GO

2 critical findings require action
```

Pour `Availability = 0`, le message actuel peut être mal compris. `0` pourrait signifier 0 % de disponibilité.

6. Utiliser la checklist seulement pour les nouveaux comptes

Le fichier possède déjà une checklist d'onboarding avec progression, premiers scans, cible uptime, première sonde, test d'alerte, Site Intelligence et triage d'un finding. security_dashboard_app

Cette checklist serait très utile, mais uniquement lorsque le compte est encore incomplet.

Je la verrais sous cette forme compacte :

```
WORKSPACE SETUP                                  2 / 6

✓ Account created
✓ First scan completed
○ Configure uptime monitoring            [Configure]
○ Test alert delivery
○ Review first finding
○ Run Site Intelligence
```

Règles UX :

- visible automatiquement sur un compte neuf ;
- rétractable ;
- mémorisée dans le navigateur ;
- cachée automatiquement lorsque les étapes essentielles sont terminées ;
- accessible ensuite depuis `Get Started`.

Ainsi, l'utilisateur expérimenté n'est pas encombré par un guide devenu inutile.

---

7. Simplifier l’aide contextuelle de la homepage

Le panneau :

```
Need help with this page?
Dashboard cockpit
```

est utile, mais il prend une ligne entière alors que la homepage est déjà très guidée.

Je le transformerais en un bouton discret :

```
? How to use this dashboard
```

Au clic, il ouvrirait le même contenu détaillé dans un drawer ou une modal.

L’aide doit rester disponible, mais ne pas concurrencer l’action principale.

---

8. Améliorer la sidebar pour un nouvel utilisateur

Dans tes captures, la sidebar est repliée. Pour une personne qui arrive pour la première fois, les symboles seuls sont difficiles à interpréter :

```
K
T
!
↯
P
C
R
✓
```

Je recommande :

- sidebar **ouverte par défaut lors des premières visites** ;
- fermeture automatique ou mémorisée uniquement après action volontaire ;
- tooltips plus descriptifs en mode réduit ;
- indicateur coloré sur la rubrique correspondant à l'étape courante ;
- badges uniquement lorsque quelque chose nécessite réellement une action.

Exemple :

```
Findings       5
Remediation    2
Incidents      0
```

Le chiffre doit représenter un travail concret, pas simplement le nombre total d'éléments historiques.

Security Posture Score

Recent Activity

- Scan on example.com completed - 2 mins ago
  2 mins ago
- Scan on example.com completed - 2 mins ago
  4 mins ago
- Scan on example.com completed - 3 mins ago
  6 mins ago
- Scan on example.com completed - 6 mins ago
  8 mins ago

Start

[+] LAUNCH NEW SCAN

Current Job Queue

RUNNING (0/2)

QUEUED (0)

AVAILABLE SLOTS (2)

Your Account

0 Active • 0 Queued

Security Tools Catalog

VULNERABILITY	ANALYSIS		MANAGEMENT
Runs	Summary	Risks	Knowledge Base
Findings	Timeline	Report	Remediation
Attack Surface	Portfolio		Compare
Compliance	Report		

Display Mode

Dark | Light | Mac/os

Live Execution

INFO

Jobs show and completed orchestrator tasks. Open a a jobow live logs and then generated run.

Need Help

HELP

Need help with this page? Scan jobs and live control.

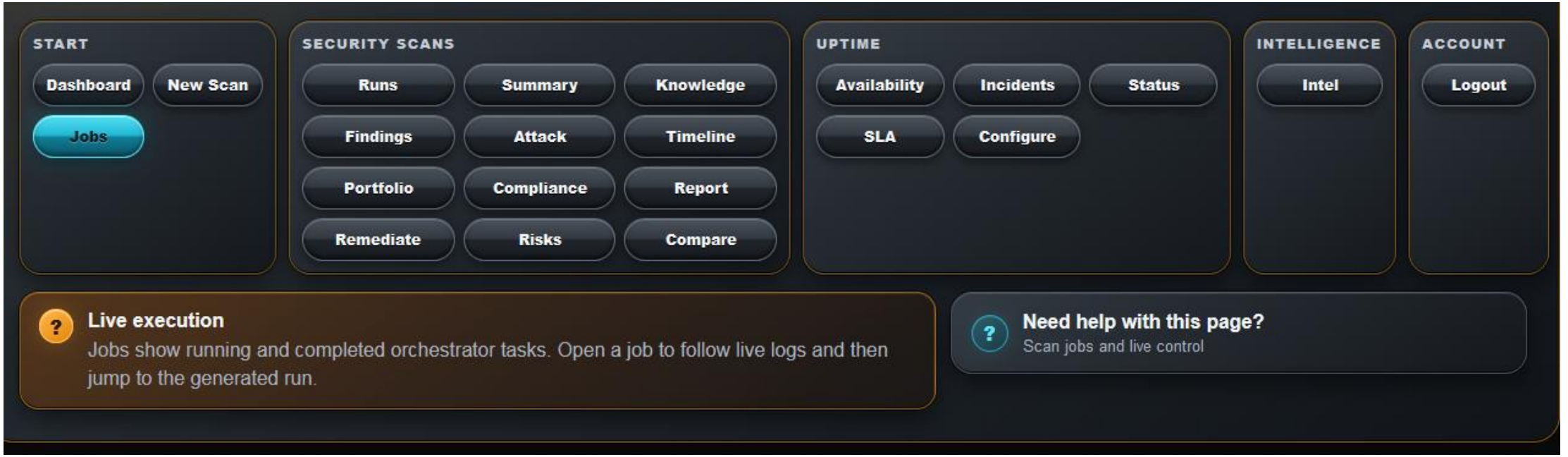
Latest Scan Results: example.com

SEVERITY	FINDING TITLE	IMPACT	RECOMMENDATION	STATUS		
!	Security Findings at Innce...	Impact entrace	Recommendation in mane...	OPEN	Details	Remediate
!	Security Pythora Finding ...	Impact entrace	Recommendation in mane...	FIXED	Details	Remediate
✕	Security Attack Parage	Impact entrace	Recommendation in mane...	IN-PROGRESS	Details	Remediate
!	Security Mane Surfaco's i...	Impact entrace	Recommendation in mane...	IN-PROGRESS	Details	Remediate

TO DO LIST

RENDRE VISUEL LA PROGRESSION

1) *Ne rendre les boutons cliquable ou visible, si la progression le permet*



ORCHESTRATOR NEW FEATURES 2026

BY IDEO LAB

AUGUST 2026

ROADMAP DASNBOARD AOUT

1. SEC-CORE-001H — Finding-to-Asset Correlation

C'est la prochaine brique backend.

Aujourd'hui :

```
Finding
   ↓
host
```

001H permettra :

```
XSS finding
   ↓
application
   ↓
endpoint

TLS finding
   ↓
service :443
   ↓
certificate

SSH finding
   ↓
IP
 ↓
port :7755
 ↓
SSH service
```

Puis : SEC-UI-002A — Asset Inventory

Nouvelle entrée majeure dans la sidebar :

```
ASSETS
```

avec une page ressemblant conceptuellement à :

```
ATTACK SURFACE / ASSET INVENTORY

105 Assets

┌────────┬──────────────┬──────┬──────────┬────────────┬─────────────┐
│ Asset  │ Type         │ Risk │ Status   │ Confidence │ Last Seen   │
├────────┼──────────────┼──────┼──────────┼────────────┼─────────────┤
│ ...    │ domain       │ 42   │ ACTIVE   │ CONFIRMED  │ 2 min ago   │
│ ...    │ IP           │ 31   │ ACTIVE   │ STRONG     │ 2 min ago   │
│ :7755  │ service/SSH  │ 48   │ ACTIVE   │ OBSERVED   │ 2 min ago   │
└────────┴──────────────┴──────┴──────────┴────────────┴─────────────┘
```

Filtres :

```
Type
Risk
Lifecycle
Criticality
Environment
Owner
Analyzer source
Correlation level
```

Et statistiques :

SEC-UI-002B — Asset 360°

Click sur :

```
3.72.171.177
```

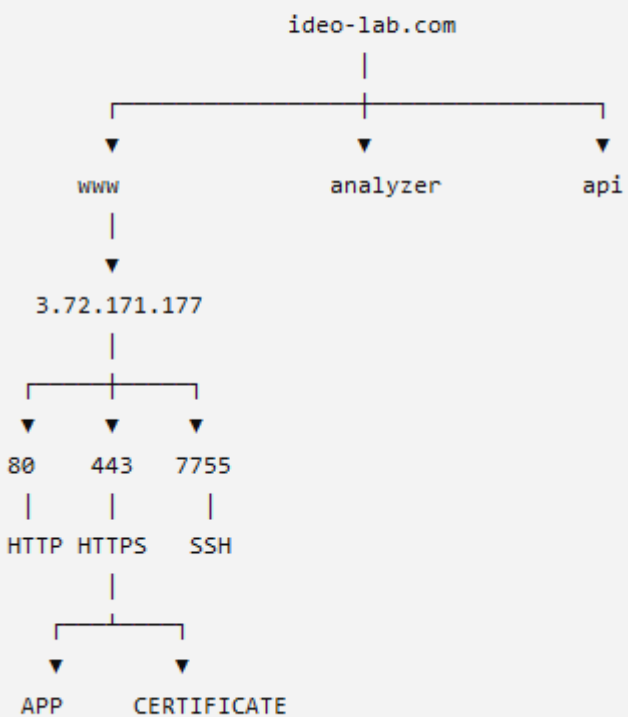
et on ouvre une fiche :

```
ASSET 360°

3.72.171.177
IP ADDRESS

Risk
36 MEDIUM

Criticality
HIGH

Environment
PRODUCTION

Lifecycle
ACTIVE

Correlation
95 CONFIRMED
```

SEC-UI-002C — Attack Surface Graph

On exploite enfin réellement `security_asset_relation`, qui existe déjà dans notre backend et porte les relations source/target typées. security_asset_graph

On pourra représenter :



Click sur un node → Asset 360°.

SEC-UI-002D — Attack Surface Changes

Là on expose 001C/001D.

Page :

```
WHAT CHANGED?
```

avec :

```
Since previous scan

+ 3 NEW
~ 2 CHANGED
- 1 DISAPPEARED
↻ 1 REAPPEARED
```

Puis :

```
NEW

api.ideo-lab.com
TCP/8443
new certificate

CHANGED

www.ideo-lab.com
TLS configuration changed

DISAPPEARED

3.72.171.177:45056
unknown service
```

SEC-UI-002E — Risk Center

On expose 001E.

Pas seulement une jauge globale.

Une vraie matrice :

```
ASSET RISK

CRITICAL     0
HIGH         8
MEDIUM      23
LOW         71
INFO         3
```

Puis :

```
TOP RISK ASSETS

Asset                  Risk     Why?
-------------------------------------------------
api...                  82      Internet + finding
SSH :7755               61      exposed admin service
www...                   54      findings + changed
```

Click sur `Why?` :

```
Risk breakdown

Exposure        +16
```

SEC-UI-002F — Ownership & Criticality

Expose 001F :

```
OWNER
Infrastructure

ENVIRONMENT
Production

BUSINESS ROLE
Administrative service

CRITICALITY
HIGH

SOURCE
Automatic

CONFIDENCE
Medium
```

et surtout bouton :

```
Edit classification
```

pour utiliser les overrides manuels que nous avons déjà construits.

SEC-UI-002G — Correlation Center

Expose 001G :

```
CORRELATION

95 / 100
CONFIRMED

Evidence sources
✓ services
✓ subdomains
✓ dns-security
✓ nuclei

Observed across
4 scan runs

Graph relations
7
```

Cela donne enfin une notion essentielle :

> « À quel point pouvons-nous faire confiance à cette information ? »

indépendamment de :

> « À quel point cet asset est-il dangereux ? »

```
PHASE 3 — ADVANCED ANALYZERS

SEC-AN-020  Authentication Surface
SEC-AN-021  OAuth / OIDC
SEC-AN-022  Session & Cookie Deep Analyzer
SEC-AN-023  Advanced CORS
SEC-AN-024  Cache Security
SEC-AN-025  Redirect Security
SEC-AN-026  Upload / Download Surface
SEC-AN-027  Sensitive Artifact Discovery
SEC-AN-028  Information Leakage
SEC-AN-029  HTTP Method Behavior
SEC-AN-030  Virtual Host / Reverse Proxy
SEC-AN-031  Advanced TLS
SEC-AN-032  Supply Chain
SEC-AN-033  Sensitive Forms
SEC-AN-034  Client-Side Injection
SEC-AN-035  postMessage
SEC-AN-036  WebSocket
SEC-AN-037  Browser Storage
```