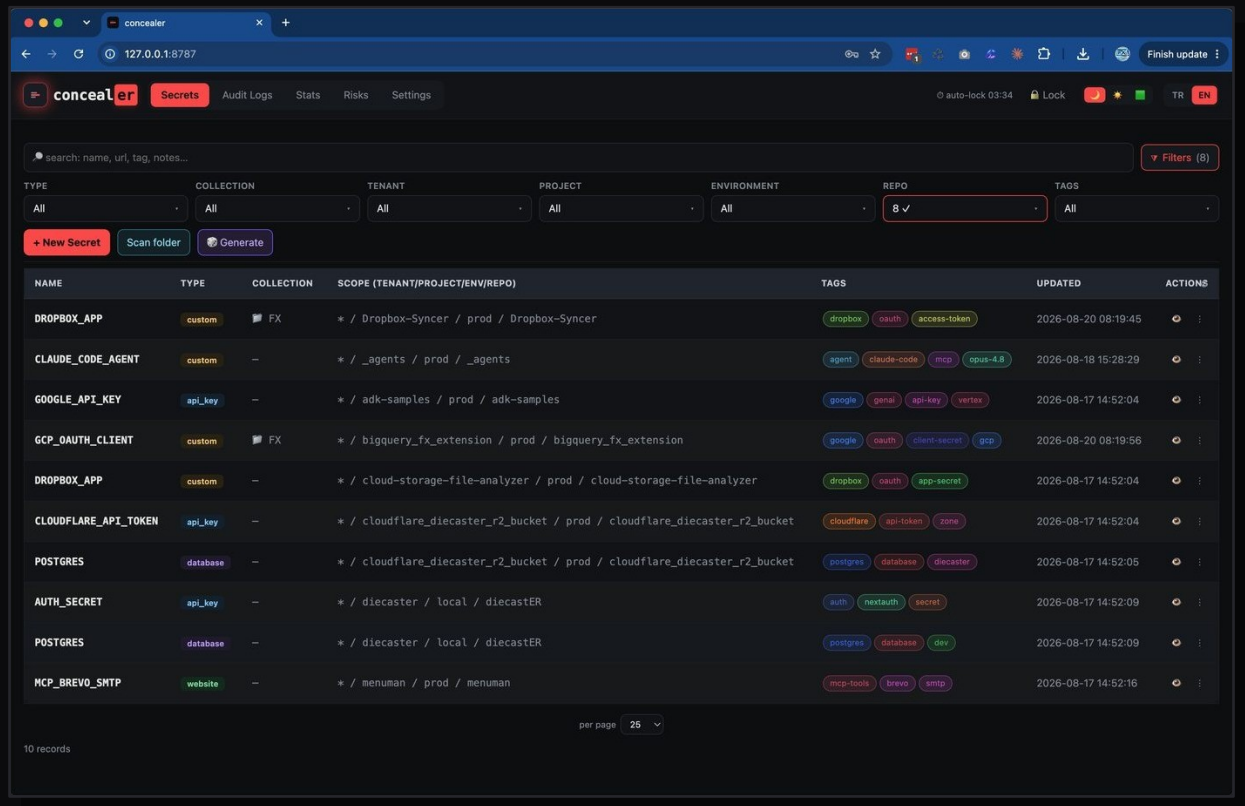
conceal er

THE AI SECRET MANAGER

Anahtarlarınızı sohbet penceresine yapıştırmayı bırakın.

Ajanlarınızın gizli anahtarlara ihtiyacı var; gerçek anahtarı onlara vermek ise sızıntının ta kendisi. concealer her şeyi kendi diskinizdeki tek bir şifreli dosyada tutar — altında SOPS ve age vardır — ve yapay zekânızın çağırabileceği MCP sunucusu dâhil her arayüze, anahtarın değerini hiç görmeden kullanma yolu verir.

CLI · WEB UI · TUI · MCP SUNUCU · CHROME



Her anahtar kapsamlı ve etiketli — ve hiçbiri bu makineden çıkmıyor

concealer.fxerkan.com

MIT · macOS · Linux · Windows

Ve size çoktan sızmış olanı da söyler.

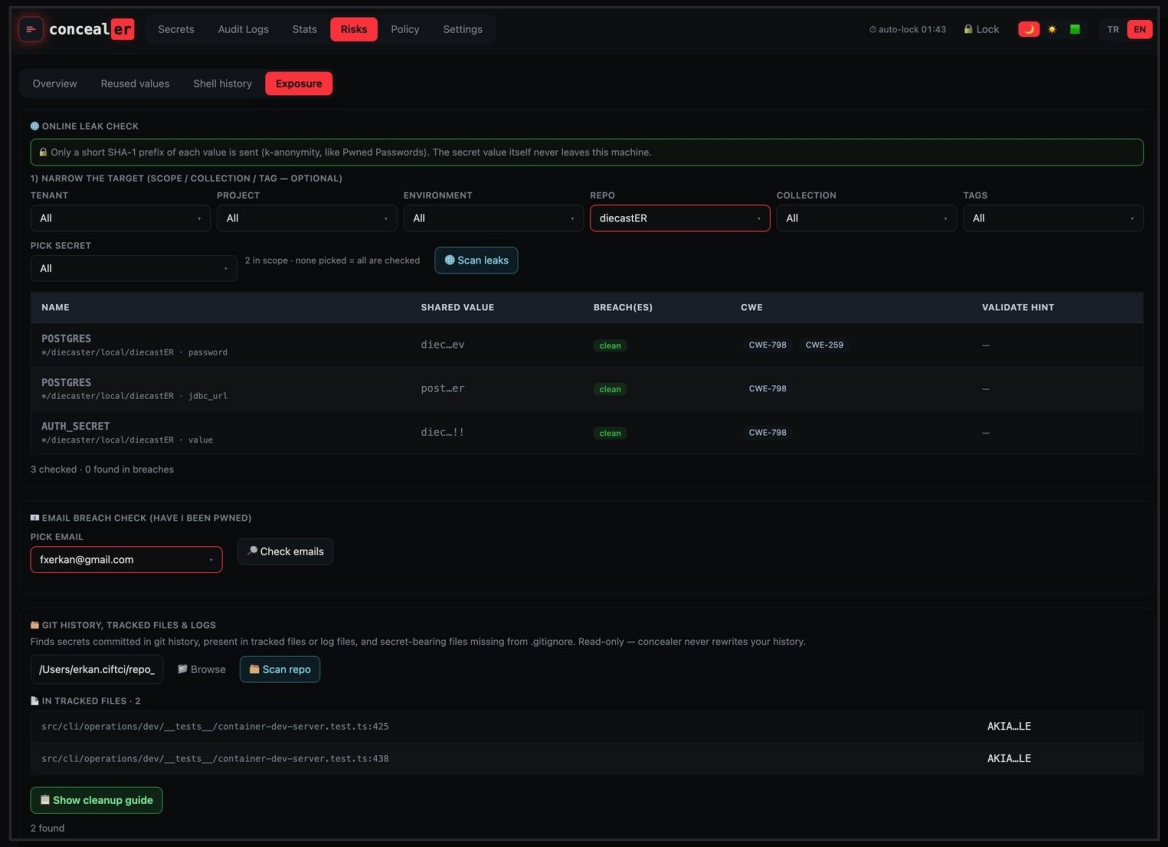
Döndürdüğünüz bir anahtar güvendedir; geçen yılın commit'inde duran anahtar değildir. concealer çalışma dizinini, git geçmişini, .env dosyalarını ve içe aktarımları tarar; hâlihazırda açıkta olanı kapatabileceğiniz bir uyarı olarak değil, maruziyet olarak listeler.

```
$ cer scan ./repo --history --envvars --import
```

```
$ cer expose
```

```
$ cer gitscan
```

Aynı tarama web arayüzünden de çalışır ve her döndürme, oynanmaya karşı korumalı denetim zincirine yazılır.



Kaynak: docs/security.md

Riskler → Maruziyet

Kodlama asistanın repo'daki her dosyayı okuyor.

Claude Code, Codex, Gemini CLI, opencode, Cursor — hepsi harika ve hepsi proje dosyalarını okuyor. Bir anahtar .env'e, credentials.json'a ya da bir sohbet penceresine düştüğü anda dışarı çıkmanın dört yolu var.

1. **Ajan tarafından okunur** ve bir model sağlayıcısına gönderilir.
2. Kullandığın aracın **log'larına veya telemetrisine kaydedilir**.
3. Kazara **git'e commit edilir** — ve geçmişte kalır.
4. Aynı isimle **on iki repo'ya kopyalanır**, hangisinin hangisi olduğu ayırt edilemez.

Bunun bedeli

4	1	0	∞
ANAHTARIN KAÇIŞ YOLU	PIŞMAN OLACAĞIN YAPIŞTIRMA	BUNU TRANSKRIPTTE DURDURAN ARAÇ	DIŞARI ÇIKTIKTAN SONRAKI ERIŞIM

- Bu listedeki hiçbir şey ajanın hatası değil — dosyalarını okumak *tam olarak bu* demek.
- Dolayısıyla çözüm "dikkatli ol" olamaz. Çözüm, değerin dosyada, transkriptte ve geçmişte hiç bulunmaması olmak zorunda.
- Bir sohbet penceresine secret yapıştırıp bir saniye sonra pişman olduysan — concealer tam bu kaşıntıyı kaşıyor.

Kaynak: README.md

Doppler, Infisical, Vault ve 1Password başka bir problemi çözüyor.

Onlar filolar, ekipler ve senkronizasyon için kurulmuş. Tek laptop'ta AI ajanıyla çalışan tek bir geliştiricinin ise hiçbirinin başa yazmadığı beş şeye ihtiyacı var:

01 · Çevrimdışı

Yalnızca yerel, kanıtlanabilir

Secret'lar makineden hiç çıkmaz. Hesap yok, senkron sunucusu yok, telemetri yok — önüne bir firewall koyup hiçbir şeyin dışarı konuşmadığını izleyebilirsin.

02 · Taşınabilir

Parolaya bağlı, donanıma değil

Kasa, yalnızca ana parolayla herhangi bir makinede açılır. Bu laptop'ın Keychain'ine ya da TPM'ine bağlı değil. Dosyaları kopyala, parolayı yaz, bitti.

03 · Git

İncelenebilir, versiyonlanabilir

Şifreli kasa, commit edebileceğin düz bir SOPS dosyası — anahtarlar görünür, değerler şifreli. Aracın kendisi kara kutu değil, okunabilir tek bir Python script'i.

04 · AI-güvenli

Ajan kullanır, görmez

MCP sunucusu ajanın isimleri listelemesine ve değerleri bir komutun ortamına enjekte etmesine izin verir. Açık metin, ajanın geri okuduğu her şeyden temizlenir.

05 · Kapsamlı

Tek yer, birçok proje

Birçok repo'da kullanılan aynı değer, tek belirsiz bir isim yerine tenant / project / environment / repo boyutlarıyla ayrıştırılır.

06 · Dürüst

Belgelenmiş tavanlar

Her sınır saklanmak yerine yazılmış: yerel denetim anahtarı, CPython'un sıfırlanmayan belleği, tek kullanıcılı web sunucusu. Bir güvenlik değerlendirmesi bunları okuyabilir.

Kaynak: README.md

Savaşta sınanmış iki aracın üzerinde ince ve denetlenebilir bir sarmalayıcı.

concealer kendi kriptografisini yazmaz. Kasayı SOPS şifreler, arka uç age'dir. concealer yalnızca eksik olan kısmı ekler: tipli secret'lar, dört boyutlu kapsam, etiketler, profesyonel bir yerel web konsolu, kurcalama-belirtir bir denetim zinciri ve bir MCP sunucusu.

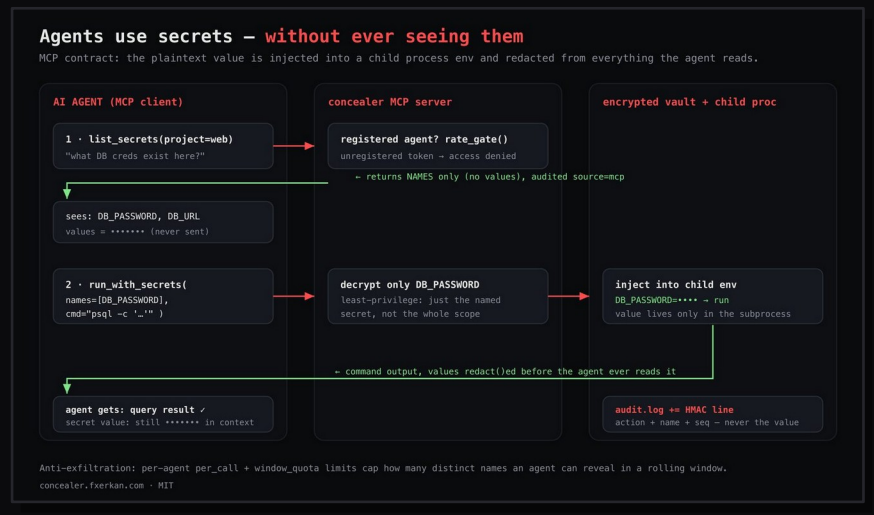
- **Tek şifreli dosya.** `secrets.enc.yaml` — age X25519 üzerinde değer başına AES-256-GCM, git dostu.
- **Tek okunabilir script.** Python 3, yalnızca standart kütüphane. MIT lisanslı. Aracın kendisi için `pip install` gerekmez.
- **Beş arayüz, tek çekirdek.** CLI · Web Arayüzü · TUI · MCP · Chrome eklentisi aynı kod yoluna — ve aynı denetim kaydına — akar.
- **Üç platform.** macOS, Linux ve Windows; her biri CI'da doğrulanıyor.
- **Anahtar diskte açık değil.** age özel anahtarı hiçbir zaman düz metin olarak diske yazılmaz — yalnızca sarmalanmış kopyaları vardır.

Kaynak: README.md

Ajanlar secret'ları kullanır. Asla okumaz.

Bir ajanı kaydet, ona iptal edilebilir bir token ver; artık canlı bir credential ile psql veya curl çalıştırabilir — değer ise bağlamına hiç girmez.

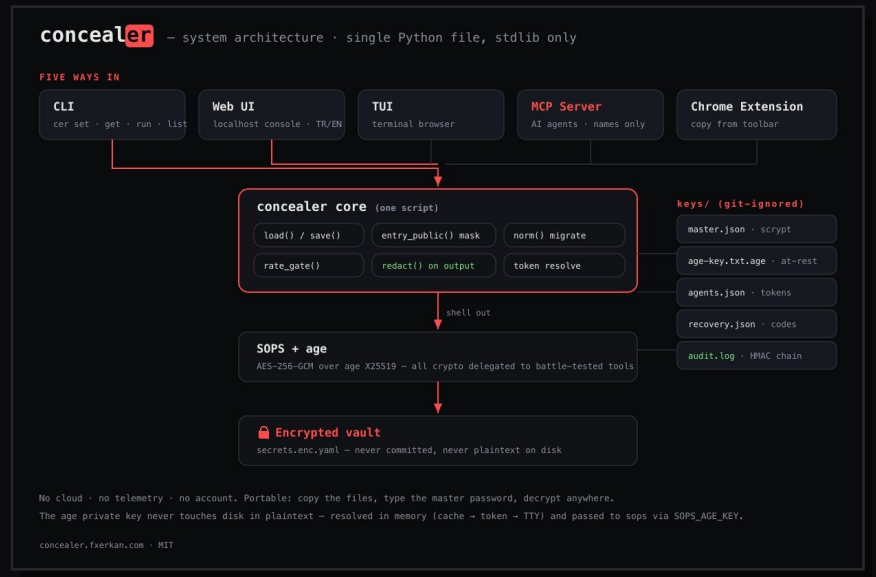
- **list_secrets** / **search_secrets** — isim, tip, kapsam ve etiket. **Asla değer değil.**
- **run_with_secrets** — eşleşen secret'lar alt ortama enjekte edilerek komutu çalıştırır; değerler **dönen çıktıdan temizlenir**.
- **set_secret** — değer yazabilir, ama az önce yazdığı değeri geri okuyamaz.
- **Kayıt zorunlu.** İnsan token'ı ya da token'sızlık *erişim reddedildi* alır. Sertleştirilmiş kasada sunucu **kapalı konumda arızalanır** (fail-closed).
- **Toplu sızdırmaya karşı.** Ajan başına kota: çağrı başına 10 satır, kayan saat içinde 25 farklı isim. Kotayı 0 yaparak ajanı tamamen engelle.
- **Her çağrı denetlenir**; `source=mcp` ve aktör olarak ajanın etiketi yazılır.
- Claude Code, Codex CLI, Gemini CLI, opencode, Cursor, Cline ve Continue ile çalışır — aynı stdio sunucusu, tek satır yapılandırma.



Kaynak: docs/mcp.md

Tek script, devredilmiş kripto, beş giriş yolu.

- **Her işlem** yükle → dict'e çöz → değiştir → kaydet → yeniden şifrele; iki yönde de `sops` çağrılır.
- **age anahtarı sops'a bellekte ulaşır** — `SOPS_AGE_KEY` üzerinden, asla geçici dosyayla değil.
- **Üç şekilde sarmalanır:** ana parolayla (`age-key.txt.age`), her kurtarma koduyla ve her kilit-açma token'ıyla.
- **Parola sorma yerine token.** İnsanlar `unlock` ile ~8 saatlik TTL token alır; her ajan kendi uzun ömürlü, iptal edilebilir token'ını alır.
- **Dış bağımlılıklar:** `sops`, `age`, `expect` — Homebrew bunları senin için kurar; Windows'ta `pywinpty`.
- **Hiçbiri commit edilmez:** `keys/`, `secrets.enc.yaml` ve `.sops.yaml` git-ignore'da. Repo aracı gönderir, kasayı asla.



Kaynak: docs/concepts.md

Dört katman ve yazıya geçirilmiş her tavan.

Kripto

Devredilmiş, icat edilmemiş

age X25519 üzerinde SOPS ile AES-256-GCM. concealer'ın kendi yazdığı tek kripto, stdlib scrypt parola doğrulayıcı ve HMAC denetim zinciri.

Anahtar

Diskte düz metin anahtar yok

Yalnızca ana parola, kurtarma kodu ve token ile sarmalanmış kopyalar vardır. Çalınan bir klasör, parola veya kod olmadan işe yaramaz.

Token

İptal edilebilir, istemcide

Token değeri yalnızca CONCEALER_TOKEN'da yaşar. Kasa bir scrypt hash'i ve sarmalanmış anahtarı tutar. İptal et, o kopya ölür.

Kurtarma

Gerçek bir ikinci faktör

8 tek kullanımlık kod. Herhangi biri kasayı kurtarır — ve passwd birini tüketir, böylece çalınmış ana parola tek başına kasayı ele geçiremez.

Tehdit modeli — belgelendiği şekliyle azaltma ve artık risk

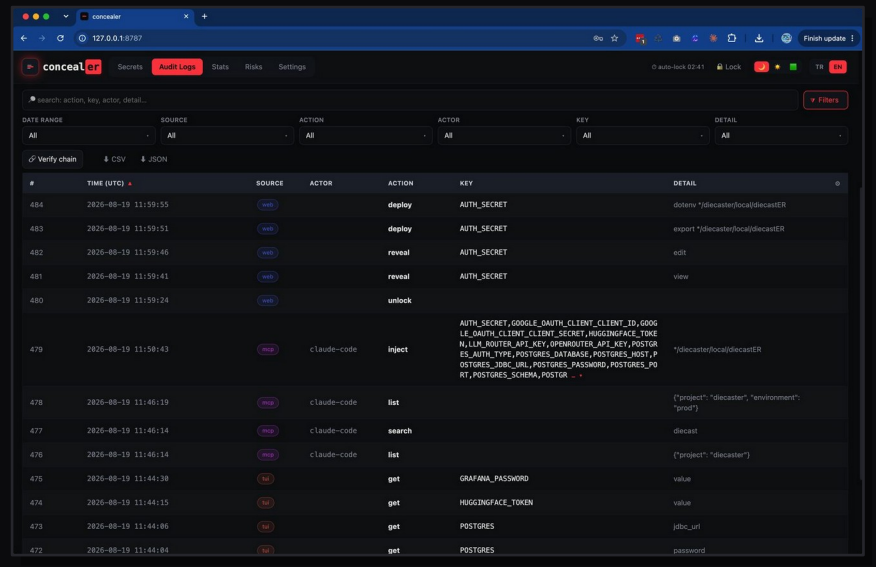
Tehdit	Azaltma	Artık risk
AI sohbetine secret yapıştırılması	MCP göstermeden enjekte eder; çıktı temizlenir	Ajan kayıtlı ve kota sınırlı olmalı
Laptop çalınır, disk kopyalanır	Anahtar diskte açık değil — yalnızca sarmalanmış kopya	Ana parolanın gücü
Ana parola sızar	`passwd` 2. faktör olarak kurtarma kodu ister	Kodlar ayrı yerde saklanmalı
Kasa kazara commit edilir	`.gitignore` kasa dosyalarını kapsar	`git add` disiplini kullanıcıda
Denetim kaydı kurcalanır	HMAC zinciri + `seq` + kuyruk çapası	Yerel anahtara erişen FS-root saldırgan zinciri yeniden üretebilir
Ajan kasayı boşaltmaya çalışır	Kayıt kapısı + ajan başına kota	Sınırlar ajan başına ayarlanabilir

Kaynak: docs/security.md

Her okuma, yazma, kopyalama ve enjeksiyon — zincirlenmiş.

Beş arayüzün tamamı için tek, yalnızca-ekleme kaydı. İsimleri ve eylemleri tutar, değerleri asla.

- **HMAC-SHA256 zincirli** — her kaydın hash'i öncekine bağlıdır.
- İmzalı yükün içinde **monoton seq**, böylece sıralama değişikliği fark edilir.
- **Kuyruk çapası** (`audit.head`) kırpmayı yakalar — son satırları silmek doğrulamayı bozar.
- **Web Arayüzünde:** eyleme / kaynağa / anahtara / tarihe göre filtrele, sayfala, satır detayını aç, zinciri doğrula, CSV veya JSON dışa aktar.
- **CLI'dan:** `cer audit` ve `cer audit verify`.
- **Açıkça belirtilen tavan:** `keys/audit.key` yereldir; dolayısıyla FS-root bir saldırgan zinciri yeniden üretebilir. Makine dışı çapa gönderimi tam yeniden üretimi yakalar; gerçek değiştirilemezlik makine dışı anahtar gerektirir.



Kaynak: docs/concepts.md

Aynı şifreli kasa, sana uyan her yoldan.

Her arayüz aynı çekirdeğe gider ve aynı denetim kaydını yazar. Hiçbiri bir atlatma yolu değil.

01

CLI

Terminalden set, get, run-with ve deploy.
Script'lenebilir, CI dostu.

02

Web Arayüzü

Yerel konsol: tip-duyarlı formlar, filtreler, deploy renderer'ları, denetim görüntüleyici.

03

TUI

Üç panel, yalnızca klavye. Terminalde gez, ara, göster ve düzenle.

04

MCP

Ajanlar isim listeler, değer enjekte eder. Yalnızca kayıtlı token, kotalı.

05

Chrome

Araç çubuğundan secret kopyala. Alan bazlı kopya, üretici, üç tema.

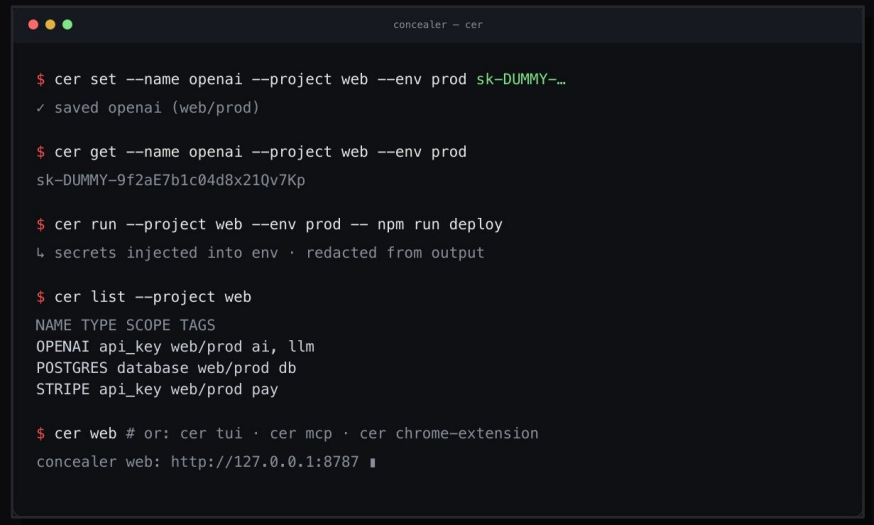
İhtiyacın…	Kullan
Script, CI, `run` veya `deploy`'a pipe etmek	CLI
Zengin formlar, filtreler, risk görünümleri, denetim görüntüleyici	Web Arayüzü
Terminalden çıkmadan hızlı klavye gezinmesi	TUI
Secret'ları görmeden kullanan bir ajan	MCP
Tarayıcıdayken bir şifre veya Wi-Fi anahtarı kopyalamak	Chrome eklentisi

Kaynak: docs/ · concealer.fxerkan.com

Tek komut, herhangi bir secret, ekrana hiçbir şey basmadan.

cer, concealer'ın kısa takma adı — ikisi de birebir aynı çalışır. Her komut bir kapsam alır; belirtilmeyen boyut joker karakterdir.

```
cer set --name OPENAI_API_KEY --project web --env prod 'sk-…' --tags ai
cer list --type database --tenant acme
cer get --name OPENAI_API_KEY --project web --env prod
cer rotate --name OPENAI_API_KEY --project web   # 32 rastgele bayt
cer run --project web --env prod npm run deploy
cer deploy --target k8s --project web --env prod
cer scan ./myrepo --history --envvars --import
cer audit verify
```

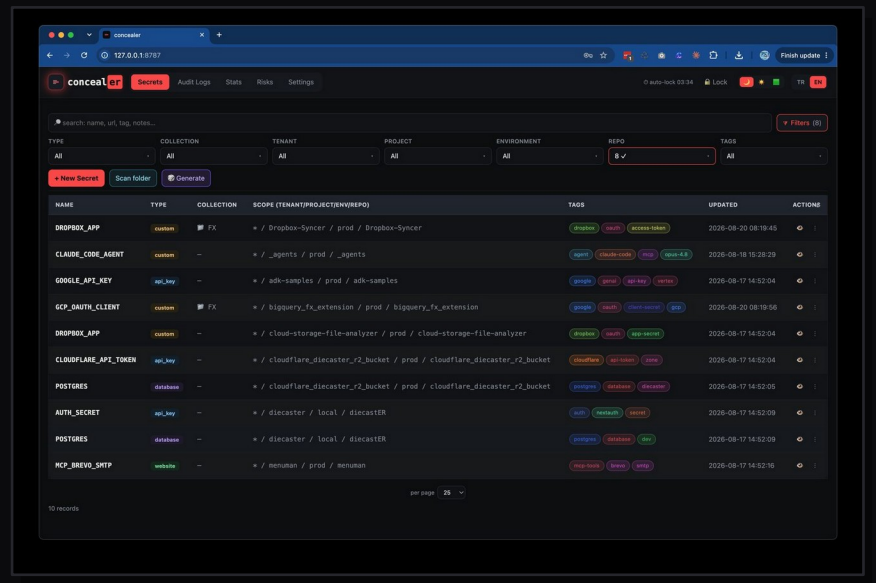


- **Kapsam otomatik algılama** — belirtmediğinde `repo` ve `project` mevcut git repo'sundan gelir.
- **Tipli enjeksiyon** — bir `database` secret'ı `PG_HOST`, `PG_PORT`, `PG_PASSWORD`… olur; `api_key` tek değişken olur.
- **8 deploy hedefi** — `dotenv` · `export` · `docker` · `json` · `k8s` · `aws-secrets` · `aws-ssm` · `github`. stdout'a render edilir; senin yerine hiçbir yere push edilmez.
- **Emniyet kilidi** — `get`, `rm` ve `rotate` belirsiz kapsamı tahmin etmek yerine reddeder.

Kaynak: docs/cli-reference.md

localhost'tan hiç çıkmayan profesyonel bir konsol.

cer web, tek sayfalık uygulamayı ve JSON API'yi yalnızca 127.0.0.1 üzerinde sunar. Tek kullanıcılı, ana parolayla açılır.



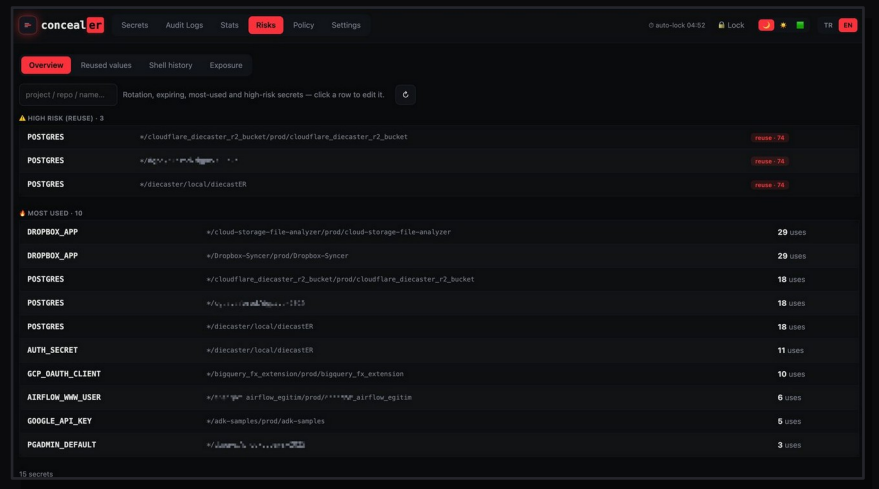
- Baştan sona **çift dilli TR / EN**; Dark · Light · Matrix temaları ve telefon/tablet uyumlu düzen.
- **Tip-duyarlı CRUD** — 15 tipin her biri tam olarak ihtiyaç duyduğu alanları gösterir; secret alanlar maskeli saklanır.
- Tip, tenant, proje, ortam, repo ve etiket için **aranabilir çok seçimli filtreler**; sıralanabilir ve yeri değiştirilebilir kolonlar — her özel alan kendi kolonu olabilir.
- **Secret başına Deploy** — sekiz hedef için tam komutu veya manifest'i doğrudan satırdan render et.
- **Otomatik temizlenen pano kopyası** (20 sn), göster/gizle anahtarı ve sert **boşta kilitlenme** (varsayılan 300 sn).
- **Riskler · Politika · Klasör tarama · Denetim · Ayarlar** birinci sınıf sekmeler olarak.

Kaynak: docs/web-ui.md

Zayıf, bayat, tekrar kullanılan ve çoktan sızmış olanı bul.

Dört alt görünüm — ve hiçbiri bir secret değerini hiçbir yere göndermiyor.

- **Genel** — Sağlık panosu: süresi geçmiş, yaklaşan, rotasyonu gecikmiş, yüksek riskli tekrar kullanım, en çok kullanılan — her biri ne kadar geciktiğiyle birlikte.
- **Tekrar** — Aynı değeri birden fazla kayıtta bulur ve patlama yarıçapını puanlar. Sızarsa, o değeri paylaşan her şey aynı anda açığa çıkar.
- **Geçmiş** — `bash` / `zsh` / `fish` geçmişini açık yazılmış credential'lar için tarar. İçe aktar, sonra satırları temizle.
- **Maruziyet** — **k-anonimlik** ile HIBP Pwned Passwords — SHA-1'in yalnızca ilk 5 hex karakteri makineden çıkar. Ayrıca e-posta ihlal sorgusu ve git geçmişi / takip edilen dosyalar / log taraması; üretilen bir temizlik kılavuzuyla.

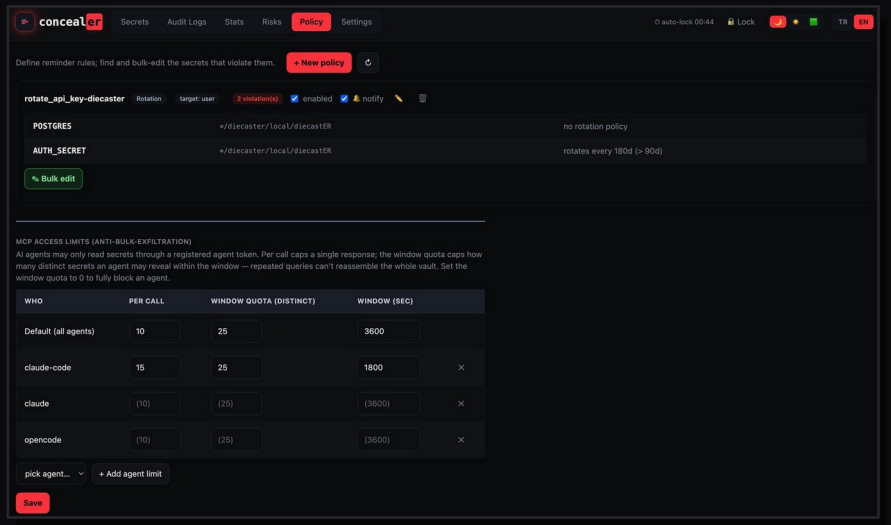


Kaynak: docs/risks.md

Senin kuralların, uygulanıyor — ve ajan kapısı.

Hatırlatma kuralları tanımla, hangi secret'ların ihlal ettiğini tam olarak gör ve ihlal edenleri toplu düzelt. Her kural bir hedef kitle taşır: user · agent · cli · web · tui · mcp · all.

Kural türü	Şu durumda işaretler…
Rotasyon	rotasyon politikası yoksa, gecikmişse veya aralığı belirlediğin üst sınırı aşıyorsa
Süre	süresi geçmişse, N gün içinde doluyorsa veya süre alanı eksikse
Tekrar	değeri başka bir kayıtla paylaşılıyorsa
İsimlendirme	adı verdiğin düzenli ifadeye uymuyorsa
Etiketleme	zorunlu tuttuğun etiketler eksikse



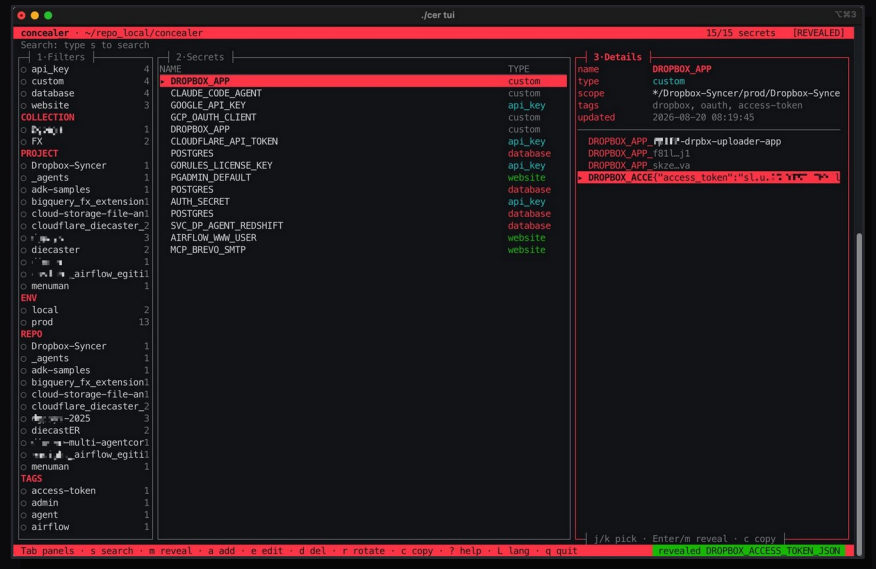

```
cer policy list
cer policy check   # ihlalde sıfırdan farklı çıkış — cron ve CI dostu
```

Kaynak: docs/policy.md

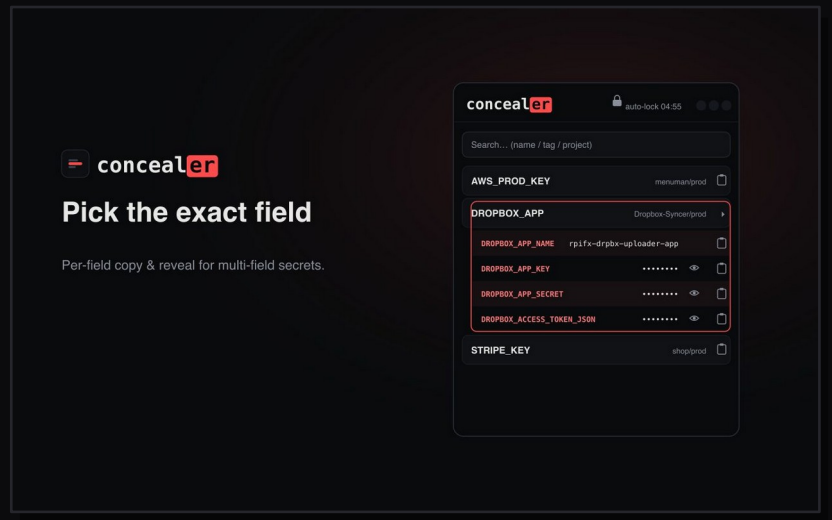
Tüm kasa, yalnızca klavyeyle.

Üç panel — Filtreler, Secret'lar, Detaylar. Shell'den çıkmadan gez, ara, göster, kopyala ve düzenle.

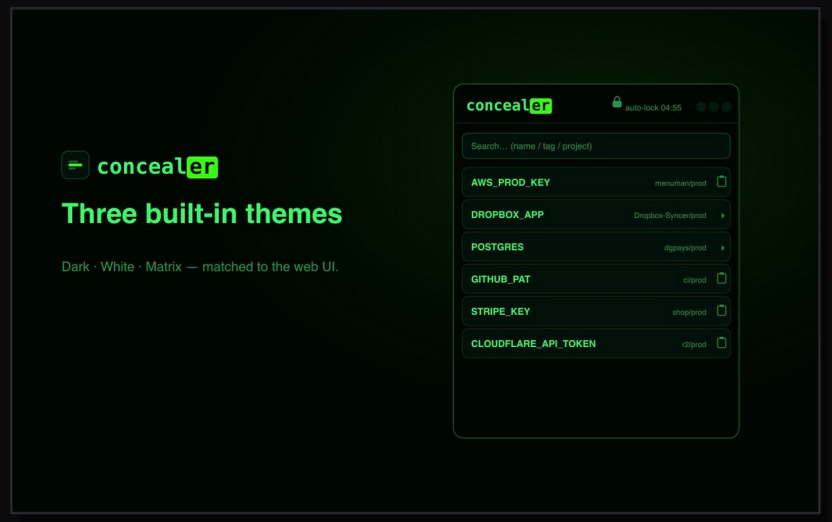
- **Vim tarzı gezinme** — `j`/`k`, `g`/`G`, paneller arası `Tab`, `1`/`2`/`3` ile atlama.
- `s` veya `/` ile **canlı arama**; `x` tüm filtreleri temizler.
- **Alan bazlı gösterme** — kayıtta `m`, ya da Detaylar'da tek bir alan seç. `c` / `u` / `w` ile değer, kullanıcı adı veya url kopyala.
- **Pano 45 saniye sonra otomatik temizlenir.** Yapıştırma yok — değer girmek için ekle/düzenle kullan.
- **Çift dilli** — `L` TR/EN değiştirir, `?` kısayol tablosunu uygulama içinde gösterir.
- **Atlatma yolu değil** — her gösterme, oluşturma, güncelleme ve silme kaynağıyla birlikte denetlenir.



Kaynak: docs/tui.md



Çok alanlı secret'lar için alan bazlı kopyalama.



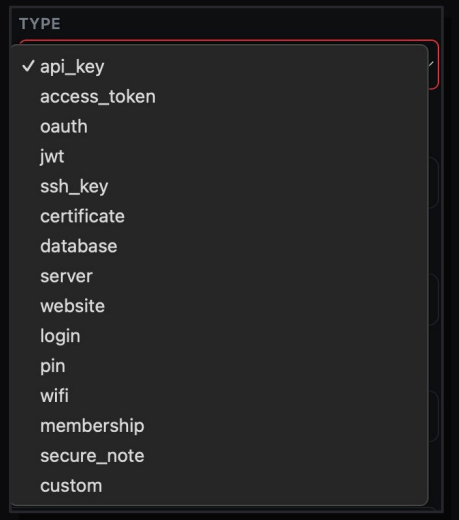
Matrix teması.

Kaynak: docs/chrome-extension.md

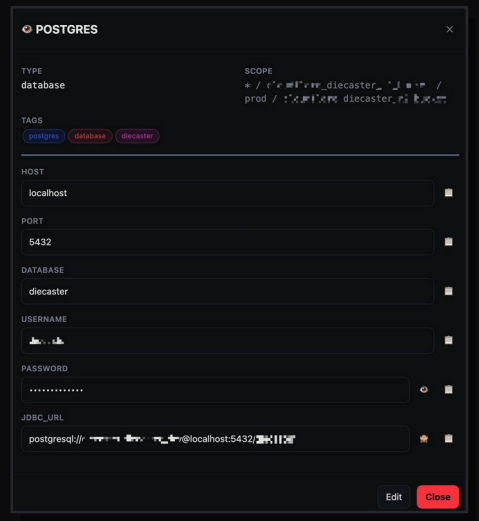
Araç çubuğundan secret kopyala.

Yerel kasanın üzerinde ince bir açılır pencere. Yalnızca 127.0.0.1 ile konuşur — bulut yok, hesap yok, telemetri yok. Chrome Web Store'da yayında.

- **Alan bazlı kopyalama** — çok alanlı secret'lar alt satırlara açılır; tam istediğin alanı kopyala veya göster.
- **İhtiyaç anında sunucu** — açılır pencere `concealer web`'i başlatır ve 15 dakika boştalıktan sonra kendini kapatır. Arkada bir şey kalmaz.
- Başlıkta **otomatik kilitlenme geri sayımı**, süre azalırken yanıp söner; pano otomatik temizlenir; değerler asla loglanmaz.
- 🎲 **Üretici** — güçlü şifre, hex, base64url, UUID. Kilidi açmadan önce de çalışır.
- **Üç tema** — Dark · White · Matrix; web konsoluyla uyumlu.
- **Yerel yardımcı concealer'ın içinde geliyor** (`concealer native-host`) — `cer chrome-extension` onu yalnızca kaydeder, makine başına bir kez.



Tip seçici.



Bir database secret'ının formu.

Tipli secret'lar, dört boyutlu kapsam.

Bir credential bir string değildir. Her tip tam olarak ihtiyaç duyduğu girdileri gösterir; secret alanlar maskeli saklanır ve yalnızca talep üzerine — denetlenerek — gösterilir.

- **15 tip** — `api_key` · `access_token` · `oauth` · `jwt` · `ssh_key` · `certificate` · `database` · `server` · `website` · `login` · `pin` · `wifi` · `membership` · `secure_note` · `custom`.
- **Maskeleme kayıt-duyarlıdır**, şu sırayla çözülür: alan bazlı geçersiz kılma → tip şablonu → alan adı sezgisi → değer sezgisi (`scheme://user:pass@host` biçimli bir DSN, düz bir alanda bile maskelenir).
- **Kapsam: tenant / project / environment / repo.** Boş boyut joker karakterdir; enjeksiyonda en özgül eşleşme kazanır.
- **Tasarımı gereği PII yok** — kredi kartı, pasaport veya kimlik numarası tipleri bilinçli olarak bulunmuyor. Bu bir makine ve hesap credential kasası.

Kaynak: docs/secret-types.md

Persona 01

DevOps / Platform Engineer

Pipeline'ların ve patlama yarıçapının sahibi. Secret'ların .env dosyalarından, shell geçmişinden ve CI log'larından çıkmasını — ve geri sızdıklarında build'i kıran bir kapı — istiyor.

Bugünkü acı

- On iki repo, her birinde kendi **.env**'i, hepsinde `OPENAI_API_KEY` adında bir anahtar ve hangisinin hangisi olduğunu ayırt etmenin yolu yok.
- **export AWS_SECRET_ACCESS_KEY=…** satırı `~/.zsh_history`'de açık metin olarak duruyor.
- Altı ay önce commit edilmiş bir anahtar var; hangi repo'ların hâlâ taşıdığını kimse bilmiyor.
- Hangi credential'ı kimin, ne zaman okuduğuna dair hiçbir kayıt yok.
- Yeni laptop = Slack thread'lerinde credential avı.

Ne kullanır

- **cer run** — kapsama uyan secret'ları alt process'in ortamına enjekte eder, çıktısından da temizler (redact). Shell'e hiçbir şey düşmez.
- **cer deploy --target** — `dotenv` · `export` · `docker` · `k8s` · `json` · `aws-secrets` · `aws-ssm` · `github` formatlarına render eder. Hiçbir yere push etmez; sen pipe'larsın.
- **cer scan** — bir klasörü, shell geçmişini ve canlı ortam değişkenlerini tarar; bulduklarını kaynağına göre etiketleyip içe aktarır.
- **cer policy check** — herhangi bir ihlalde sıfırdan farklı çıkış kodu döner. Cron'a, pre-commit hook'a veya CI'a koy.
- **cer gitscan** — git geçmişini ve takip edilen dosyaları kasa değerleriyle tarar, ardından bir temizlik kılavuzu yazar (geçmişi asla kendisi yeniden yazmaz).
- **cer audit verify** — HMAC zincirini ve kuyruk çapasını yeniden hesaplar.

Ne kazanır

- Değerler terminale, transkripte ya da log'a hiç ulaşmaz.
- Kapsam git repo'sundan otomatik algılanır — `--repo` ve `--project` kendini doldurur.
- Enjeksiyonda en özgül kapsam kazanır: `acme/proj-a/prod` > `proj-a` > global.
- Çalıştırılacak, yamalanacak ya da parası ödenecek sıfır altyapı.
- Kasa şifreli bir SOPS dosyası — kodun yanında diff'lenir ve versiyonlanır.

Persona 02

Cloud Architect / Solution Architect

Birçok müşteri ve üç bulut üzerinde demo, PoC ve pilot yürütüyor. Tasarımı bir güvenlik değerlendirmesinde savunmak — ve sunucu ayağa kaldırmadan her şeyi devretmek — zorunda.

Bugünkü acı

- PoC için alınan müşteri credential'ları, kişisel ve kurum içi anahtarlarla iç içe.
- Üç bulut, on iki tenant, dört ortam — tek düzlemsel bir isim alanı bunu ifade edemez.
- Güvenlik değerlendirmesi "bu credential nereye gidiyor?" diye soruyor ve dürüst cevap omuz silkmek.
- İki haftalık bir PoC için Vault kurmak saçma; SaaS kasa ise satın alma süreci ve DPA demek.
- Kasa tek bir laptop'ın Keychain'ine bağlı, dolayısıyla devir teslim aslında yeniden kurulum.

Ne kullanır

- Taksonomi olarak **tenant / project / environment / repo** — aynı `DATABASE_URL` her müşteri ve aşama için bir kez, isim çarpıtmadan var olur.
- Konsol olarak **Web Arayüzü**: tip-duyarlı formlar, çok seçimli filtreler, sıralanabilir kolonlar, secret başına deploy renderer'ları.
- Değerlendirme çıktısı olarak **tehdit modeli tablosu** ve **belgelenmiş tavanlar** — aracın neyi korumadığı dahil.
- **cer export / import** — müşteriye verilecek tek, parola korumalı `.cerbak` paketi.
- **Taşınabilirlik**: `secrets.enc.yaml` + `keys/` kopyala, yeni makinede ana parolayı yaz. TPM yok, Keychain yok, yeniden anahtarlama yok.
- Build-vs-buy konuşması için **karşılaştırma matrisi**.

Ne kazanır

- Sunucu yok, container yok, bulut tenant'ı yok, satın alma yok — `init` ve kasa hazır.
- Kriptografi **SOPS** (CNCF) ve **age**'e devredilmiş: age X25519 üzerinde AES-256-GCM. Ev yapımı hiçbir şey yok.
- Müşterinin güvenlik ekibinin gerçekten okuyabileceği, MIT lisanslı tek bir okunabilir Python script'i.
- Kopyalanmış bir klasör, ana parola veya kurtarma kodu olmadan **işe yaramaz**.
- Tasarımı gereği PII tipi yok — bu bir makine credential kasası, kimlik deposu değil.

`docs/security.md` · `docs/portability.md` · `docs/comparison.md`

Persona 03

End User (Son Kullanıcı)

DevOps mühendisi değil. Wi-Fi şifreleri, kapı PIN'leri, site girişleri ve birkaç API anahtarı var — ve bunların hiçbirinin not uygulamasında, tarayıcı profilinde ya da başkasının bulutunda olmasını istemiyor.

Bugünkü acı

- Şifreler bir not dosyasında, bir Excel'de ya da tarayıcının şifre yöneticisinde.
- Aynı şifre yarım düzine sitede tekrar kullanılıyor; hangilerinin sızdığı ise meçhul.
- Ayda 3–12 dolarlık bir abonelik daha ve başkasının sunucusunda bir hesap daha.
- Ana parolayı unutmak her şeyi kaybetmek anlamına geliyor.

Ne kullanır

- **Chrome eklentisi**: araç çubuğu simgesine tıkla, kilidi aç, kopyala. Çok alanlı secret'lar açılır, tam istediğin alanı kopyalarsın.
- `127.0.0.1:8787` adresindeki **Web Arayüzü** — tamamen çift dilli TR/EN; `wifi · pin · login · website · membership · secure_note` formları hazır.
- 🎲 **Üret** — güçlü şifre, hex, base64url veya UUID; tek tıkla kopyala.
- **Riskler → Maruziyet**: bir değeri HIBP Pwned Passwords'a, bir adresi HIBP ihlal kayıtlarına karşı kontrol et.
- Kurulumda basılan **8 kurtarma kodu** — herhangi biri seni geri içeri alır.

Ne kazanır

- Hesap yok, abonelik yok, senkronizasyon sunucusu yok. Hiçbir şey makineden çıkmaz.
- Pano 20 saniye sonra otomatik temizlenir; oturum boşta kalınca kilitlenir (varsayılan 300 sn).
- Sızıntı kontrolü **k-anonimlik** ile çalışır — SHA-1'in yalnızca ilk 5 hex karakteri makineden çıkar.
- Üç tema (Dark · White · Matrix) ve baştan sona Türkçe arayüz.
- Dürüst sınır: mobil uygulama ve tarayıcı otomatik doldurma yok — iş akışı kopyala butonu.

Yalnızca yerel, sıfır altyapı, ajan-yerlisi — diğer üç pazarın açık bıraktığı boşluk.

✓ var ~ kısmen × yok ★ yalnızca concealer

Yetenek	concealer	1Password / Bitwarden	HashiCorp Vault	Doppler / Infisical	SOPS + age (ham)	pass / gopass	secretctl
Dağıtım	Yerel, tek dosya	SaaS	Self-host / HCP	SaaS (veya self-host)	Yerel	Yerel	Yerel, tek ikili
Sunucu veya daemon gerekir	× yok	bulut	✓ sunucu	✓	×	×	× yok
Maliyet	Ücretsiz (MIT)	~3–12 $ / kişi / ay	Ücretsiz OSS / $$$ kurumsal	ücretli paketler	Ücretsiz	Ücretsiz	Ücretsiz
Git'te versiyonlanabilir kasa	✓ değerler şifreli	×	~	×	✓	✓	× SQLite blob
Tipli secret'lar (db / api / ssh…) ★	✓ şablonlu	~ öğe tipleri	×	×	×	×	—
Dört boyutlu kapsam	✓ tenant/proje/ortam/repo	~ kasa, etiket	✓ yollar	✓ ortamlar	×	~ dizinler	~ joker
Kurcalama-belirtir denetim kaydı	✓ HMAC + çapa	~ bulut log	✓	✓	×	~ git log	✓ HMAC
AI ajanları için MCP-yerlisi	✓ kapı + kota	×	~ SDK	~ SDK	×	×	✓ MCP
Ajan başına sızdırma kotası ★	✓	×	~ politika	×	×	×	—
Tamamen çevrimdışı çalışır	✓	~ önbellek	~	×	✓	✓	✓
Dinamik / kiralık secret	×	×	✓	~	×	×	×
Çok kullanıcı / RBAC / SSO	× tek sahip	✓	✓	✓	~ alıcılar	~ anahtarlar	×
Mobil uygulama / otomatik doldurma	×	✓	×	×	×	~	×

Kaynak: docs/comparison.md

concealer nerede kazanıyor — ve nerede başka bir şey kullanmalısın.

Kazandığı yerler

- **Sıfır altyapı.** Sunucu, container, bulut tenant'ı ya da daemon yok. `init` ve kasa hazır.
- **Git-yerlisi.** Kodla aynı repo'da diff'lenip versiyonlanan şifreli bir YAML dosyası.
- **Ajan-öncelikli.** Matristeki tek araç ki MCP sunucusu AI ajanı tehdit modelleri üzerine kurulmuş: kayıt kapısı, ajan başına sızdırma kotası, temizlenmiş çıktı.
- **Tasarımı gereği kurcalama-belirtir.** HMAC zinciri artı kuyruk kırpmayı yakalayan bir baş çapası — bulut denetim hattı gerekmez.
- **Kilitlenme yok, telemetri yok, okunacak tek dosya.** Tescilli bir bulut kasasına güvenmekle karşılaştır — 2022 LastPass ihlaline bak.

Kazanmadığı yerler — başkasını kullan

- **Ekipler.** SSO yok, RBAC yok, kullanıcı bazlı paylaşım yok. Tek sahipli bir kasa. → 1Password / Bitwarden / Vault.
- **Dinamik secret ve kiralama.** Talep üzerine üretilen kısa ömürlü DB credential'ı yok. → HashiCorp Vault.
- **Otomatik rotasyon ve CI/CD senkron dokusu.** Rotasyon manuel. → Doppler / Infisical.
- **Tüketici deneyimi.** Mobil uygulama yok, tarayıcı otomatik doldurma yok, passkey yok. → 1Password / Bitwarden / Keeper.
- **Anahtar teslim uyumluluk belgeleri.** FedRAMP veya SOC 2 yok — bunlar aracı işleten kurumun kazandığı belgelerdir, aracın içinde gelmez. → Keeper / Vault / bulut KMS.

İhtiyacın…	Kullan
Sunucusuz, git'te versiyonlanan kişisel veya tek geliştiricili kasa	**concealer**
Yerel AI ajanları ve MCP istemcileri için güvenli secret erişimi	**concealer**
Sadece bir repo içindeki config dosyasını şifrelemek	**concealer**
Ekip paylaşımı, SSO, mobil otomatik doldurma	1Password / Bitwarden
Dinamik DB credential'ı, kiralama, servis olarak şifreleme	HashiCorp Vault
CI/CD'ye senkronlanan yönetilen çok ortamlı secret'lar	Doppler / Infisical

Kaynak: docs/comparison.md

Yaklaşık bir dakikada çalışır durumda.

Paket yöneticini seç — her biri concealer'ın sarmaladığı sops ve age ikililerini de kurar. Sonra bir kez init: 8 kurtarma kodu ve bir başlangıç token'ı basar, ardından düz metin age anahtarını diskten siler.

macOS · Homebrew

```
# sops, age, expect ile birlikte gelir
brew install fxerkan/tap/concealer

concealer init
eval "$(cer unlock)"
```

Linux · pipx

```
pipx install concealer
sudo apt install sops age

concealer init
eval "$(cer unlock)"
```

Windows · Scoop

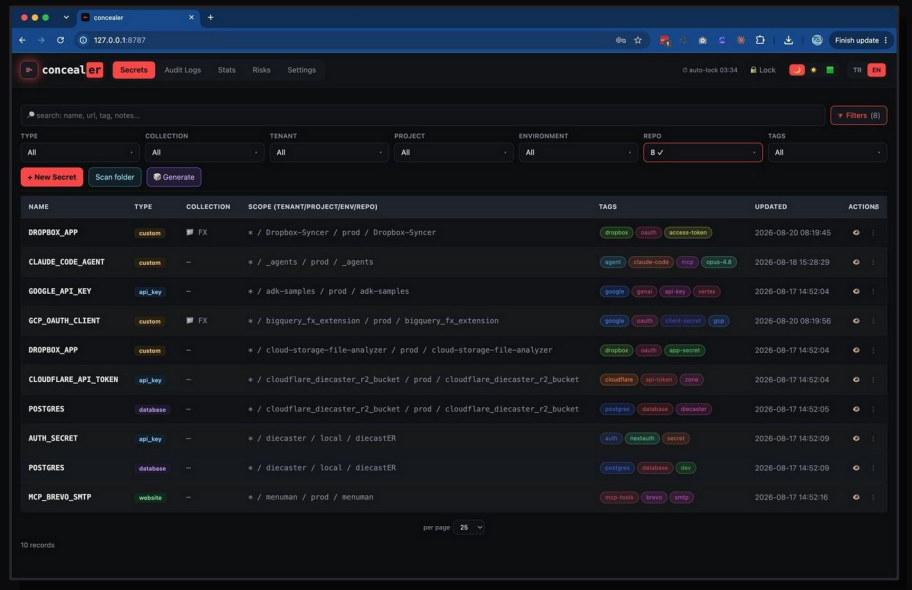
```
scoop bucket add fxerkan \
  https://github.com/fxerkan/scoop-bucket
scoop install concealer

concealer init
$env:CONCEALER_TOKEN = "…"
```

Kaynak: docs/installation.md

conceal er

Sohbet pencerelerine anahtar yapıştırmayı bırak.

Yalnızca yerel. Taşınabilir. Açık kaynak. Artık kod yazdığımız şekle göre tasarlandı.




```
brew install fxerkan/tap/concealer
```

```
pipx install concealer
```

```
scoop install concealer
```

MIT

macOS · Linux · Windows

- concealer.fxerkan.com
- GitHub
- Tam karşılaştırma
- Chrome Web Store
- Destek

---

concealer v0.9.24 · MIT · FXerkan tarafından geliştirildi — Code more, worry less.