Sentinel by CryptoITData

Ihr Server wird gerade jetzt angegriffen. Sehen Sie, von wem.

Ein Sicherheitsagent, der rund um die Uhr auf einem einzigen Linux-Server läuft: er sammelt aus sechs Quellen, eröffnet Vorfälle, legt den Blockier-Knopf auf Ihr Telefon, scannt Schwachstellen und erzeugt Patch-Prozeduren mit einem geprüften Rückweg.

Die Zahlen unten sind echt, aus einer Produktionsinstallation — AlmaLinux 9, 14 überwachte Anwendungen, die letzten 24 Stunden.

Erkennt

Sechs Sammelquellen, deterministische Regeln alle 10 Sekunden, Vorfälle per Fingerabdruck dedupliziert.

Alarmiert

Der Vorfall in Sekunden auf Telegram, mit dem Blockier-Knopf darunter und einem KI-Urteil neben dem deterministischen.

Repariert

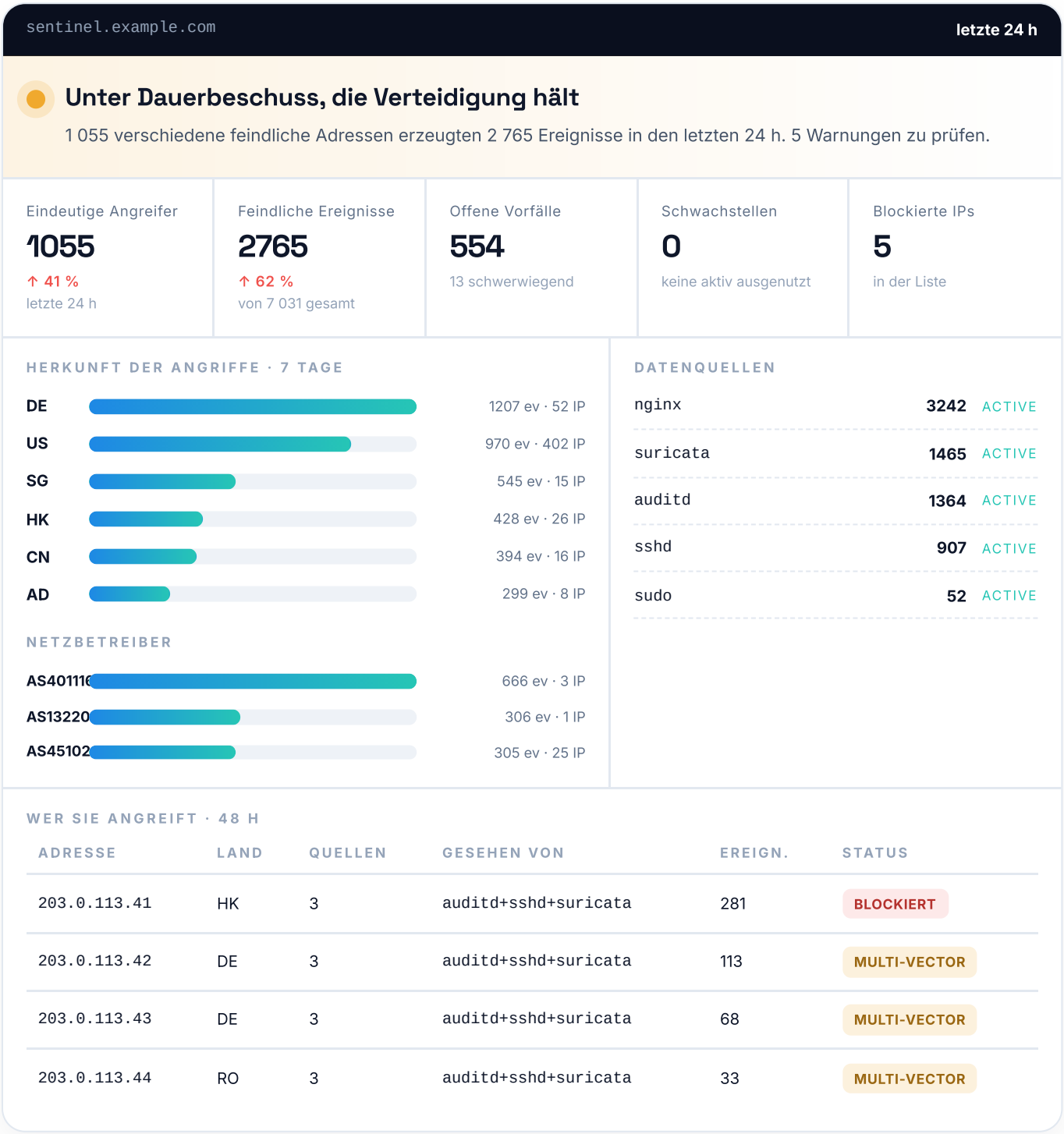
Patch-Pläne, priorisiert nach KEV und Exposition, mit geprüftem Backup und automatischem Rollback.

1 055	2 765	6	0	14
eindeutige Angreifer / 24h	feindliche Ereignisse	Sammelquellen	offene Schwachstellen	aktive Dienste

DAS SICHERHEITS-PANEL

Die erste Zeile beantwortet eine einzige Frage: bin ich okay?

Die Lesereihenfolge ist das Design. Das Urteil, dann die Zahlen dahinter, dann die Deutung, dann die Tabellen, aus denen es kommt. Wer nur die erste Zeile liest, hat schon die richtige Antwort.



Das echte Panel, mit den Produktionsdaten von heute. „Multi-Vektor“ heißt, dass die Adresse von drei unabhängigen Kollektoren gesehen wurde — ein Massenscanner berührt eine Oberfläche; wer drei probiert, hat Sie ausgewählt.

DEUTUNG

Eine Zahl ist keine Beobachtung.

„2 765 Ereignisse“ ist eine Zahl. „root wurde 1 086 Mal von 93 Adressen angegriffen — deaktivieren Sie die Passwort-Anmeldung für root“ ist eine Beobachtung. Die Engine führt deterministische Regeln über die Daten aus und liefert nur Dinge, bei denen Sie handeln können.

5 Adressen greifen Sie gleichzeitig an mehreren Fronten an

Sie tauchen in den letzten 48 h in 3 unabhängigen Quellen auf. Ein Massenscanner berührt eine Oberfläche; wer drei probiert, hat Sie ausgewählt.

```
ZU TUN /block 203.0.113.41 24h
```

root ist Ziel Nr. 1 — 1 086 Versuche von 93 Adressen

Solange die SSH-Passwortanmeldung für root möglich ist, hat jeder dieser Versuche eine Chance.

```
ZU TUN grep PermitRootLogin /etc/ssh/sshd_config
```

Konzentrierte Angriffsinfrastruktur: Netiface America, Inc.

666 feindliche Ereignisse von nur 3 Adressen im selben Netz. Das ist gemietete Angriffsinfrastruktur, keine verstreuten gekaperten Rechner — und die nächste Adresse kommt von dort.

5 Sperren stehen in der Liste, aber nicht mehr in der Firewall

Sperren überleben einen Neustart absichtlich nicht — das ist das Aussperr-Netz. Das Panel sagt es Ihnen, statt Sie glauben zu lassen, Sie seien geschützt.

Aktive Kampagne gegen GLPI: 560 Sondierungen

Hartnäckig und gezielt — aber Sie betreiben kein GLPI, also kann sie Sie nicht treffen. Hintergrundrauschen, nützlich nur, um zu wissen, in welcher Welle Sie stecken.

Null offene Sicherheitslücken

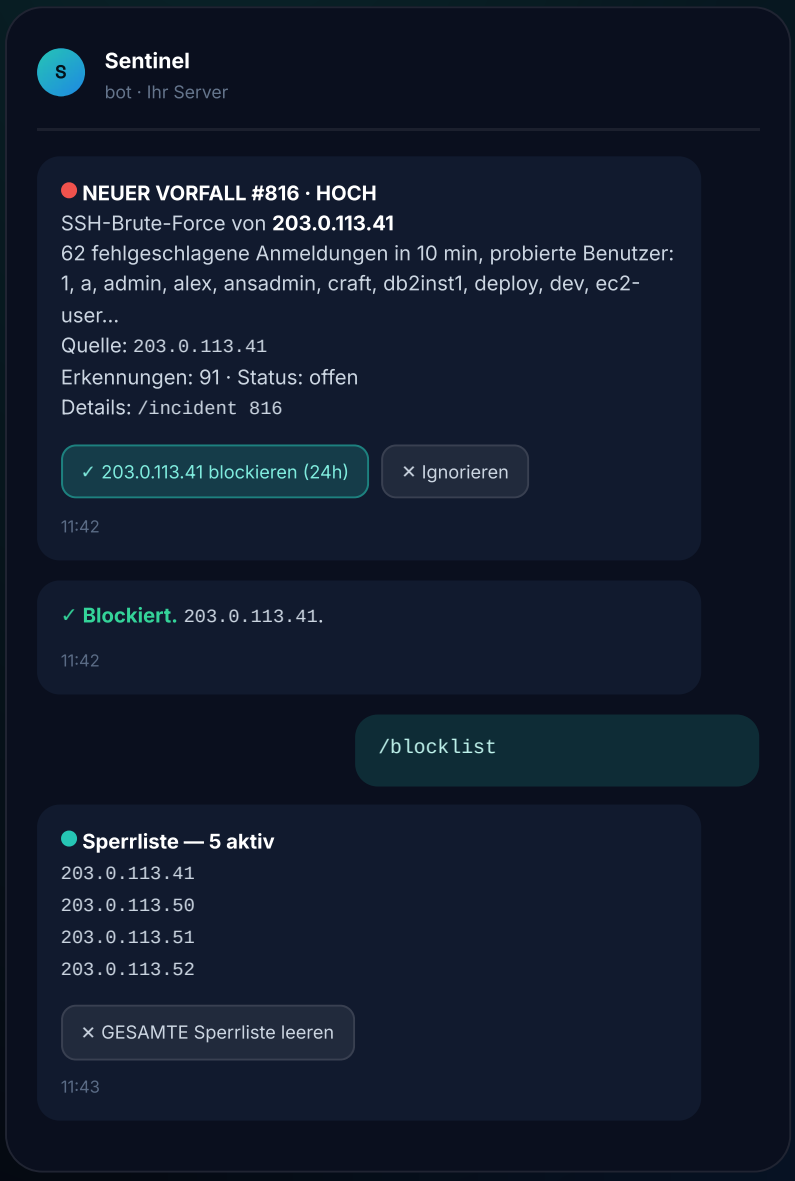
4 716 wurden behoben und vom nächsten Scan bestätigt.

Alle 11 Beobachtungen oben entstehen aus den echten Daten des Servers, ohne Aufruf eines Modells — die Seite rendert sofort und sagt zweimal hintereinander dasselbe.

DER NOTFALLKANAL

Der Alarm erreicht Ihr Telefon, mit dem Stopp-Knopf darunter.

Long-Polling, also kein zusätzlicher offener Port und kein öffentlicher Endpunkt zum Fälschen. Es funktioniert selbst, wenn nginx oder das Zertifikat kaputt sind — genau dann, wenn Sie es brauchen.



Der Knopf ist nicht die Autorität

Ein Druck darauf läuft durch dieselbe Rollenprüfung wie ein getippter Befehl, und der Executor weigert sich ohnehin, die Admin-Adresse zu sperren — auch wenn der Knopf versehentlich gedrückt wird.

Er überlebt den Ausfall

Der Bot holt die Nachrichten, statt sie zu empfangen: kein offener Port zeigt auf ihn und es gibt keinen öffentlichen Endpunkt zum Fälschen. Fallen nginx, TLS oder das Panel aus, bleibt der Kanal stehen.

Lesen ist grün, Ändern ist rot

Befehle, die etwas ändern, verlangen eine eigene Bestätigung, und die zerstörerischen eine zweite, die das Ziel noch einmal nennt.

Ein echtes Transkript: die Nachrichten, die Knopfbeschriftungen und die Befehlsnamen sind die, die der Bot sendet, keine Attrappe.

VOM PAKET ZUR ENTSCHEIDUNG

Was zwischen dem Angriff und dem Knopf auf Ihrem Telefon passiert.

Fünf Stufen, jede mit einer Aufgabe, die sie nicht an die nächste abgibt. Der deterministische Teil entscheidet; das Modell erklärt danach.

01 · SAMMLUNG

Sechs Quellen, am Kollektor gefiltert

sshd, sudo/su, nginx, auditd, Suricata. Rauschen wird vor der Datenbank verworfen — 96 000 Engine-Diagnosen pro Tag kommen nie dort an.

02 · ERKENNUNG

Deterministische Regeln, alle 10 Sekunden

Ehrliche Schwellen über indiziertem SQL: Brute-Force, Web-Enumeration, IDS-Alarme, Volumenanomalie. Null Kosten pro Ereignis.

03 · VORFALL

Per Fingerabdruck dedupliziert

400 Versuche von einer Adresse werden ein Vorfall, nicht 400. Der Akteur bekommt Land und Betreiber aus lokalen Geo-Datenbanken.

04 · KI-TRIAGE

Urteil, keine Punktzahlen

Das Modell kalibriert die Schwere neu und schreibt die Zusammenfassung. Das deterministische Urteil bleibt daneben — Widerspruch ist sichtbar.

05 · ENTSCHEIDUNG

Ihre, mit einem Tippen

Auto-Block existiert und ist getestet, wird aber **aus** ausgeliefert. Die ersten 72 h zeigen Ihnen, was er blockiert *hätte*.

× Das Modell entscheidet keine Sperren

Erkennung, Bewertung und Entscheidung sind deterministisch und laufen zu null Kosten. Eine nicht erreichbare API oder ein aufgebrauchtes Budget ändert nichts daran, was blockiert wird.

× Das Modell führt nichts aus

Der Plan, den es schreibt, läuft durch einen deterministischen Validator, der ihn ablehnt, wenn er nicht sicher ist. Das Modell verfasst, der Validator entscheidet.

× Das Modell liegt nicht auf dem kritischen Pfad

API ausgefallen oder Budget aufgebraucht: Erkennung, Blockierung und Alarmierung laufen weiter, markiert mit „KI-Analyse nicht verfügbar“.

× Das Modell überschreitet das Budget nicht

Das Limit wird vor jedem Aufruf geprüft, und die Kosten pro Vorfall stehen im Panel.

Die Erkennung läuft zu null Kosten pro Ereignis; das Modell wird nur für Vorfälle aufgerufen, die die Schwereschwelle überschreiten — typischerweise ein paar Dutzend am Tag.

SCHWACHSTELLEN UND PATCHING

Scannen ist der leichte Teil. Was Sie zuerst reparieren, ist der schwere.

Priorisierung aus **CVSS × EPSS × KEV × Exposition × Kritikalität**. Ein CVSS 6.5 auf der CISA-Liste aktiv ausgenutzter Lücken schlägt eine 9.8, die niemand ausnutzt.

sentinel.example.com/patches/3 — **Patch-Plan #3** — VALIDIERT

Risikostufe	Auswirkung	Geschätzte Ausfallzeit	Reversibel	Neustart nötig
hoch	ein Dienst	0s	ja	nein

ZIEL · HASH

ZIEL `postgresql · hash 909a1d35a818de31…`

PREFLIGHT — PRÜFUNGEN VORHER

check_disk	no_incident	curl_installed
disk_free	no_open_incident	pkg_version

EINSPIELEN · ROLLBACK

EINSPIELEN `dnf -y update curl`

ROLLBACK `dnf -y downgrade curl-7.76.1-29.el9`

BACKUP `rpm_state · curl`

× Ausgenutzt schlägt theoretisch

Eine Schwachstelle auf der CISA-Liste aktiv ausgenutzter Lücken steigt über eine höhere Punktzahl, die draußen niemand nutzt.

× Exposition schlägt Inventar

Was aus dem Internet erreichbar ist, wird vor dem gepatcht, was mit derselben CVE hinter der Firewall sitzt.

× Kritische Assets laufen nie automatisch

Datenbanken und alles, was Sie als kritisch markiert haben, bekommen nie einen automatisch erzeugten Patch — sie bekommen einen Plan, den Sie freigeben.

Ein echter Plan, automatisch erzeugt und beim ersten Versuch validiert. Der Rollback nagelt die alte Version korrekt fest — die langweilige Lösung, die funktioniert.

WIE EIN PATCH ENTSTEHT

Sechs Schritte, und das Modell ist nur beim ersten dabei.

01 **Erzeugung**

Aus den deterministischen Fakten des Scans. Das Modell bekommt keine Shell.

02 **Validierung**

Jeder Befehl ist eine Liste, das erste Element aus einer erlaubten Liste, `rm` verboten. Ungültig → ein neuer Versuch → abgelehnt.

03 **Freigabe**

Zwei Bestätigungen auf Telegram, ein Einmal-Token, an den Hash gebunden.

04 **Backup**

Ein Punkt, geprüft durch erneutes Lesen der Archive. Ohne ihn startet der Patch nicht.

05 **Anwendung**

Schritt für Schritt, *vor* jedem Befehl in die Datenbank geschrieben.

06 **Rollback**

Automatisch bei jedem Fehler. Ein gescheiterter Rollback ist ein eigener, schwererer Zustand.

> Das Modell ist bei Schritt 01 dabei und sonst nirgends. Alles danach ist deterministisch: ein Validator, der ablehnt, ein Einmal-Token, ein Backup, das vor der Anerkennung erneut gelesen wird, und ein Rollback, der bei jedem Fehler läuft.

× Das Modell bekommt keine Shell

Es schreibt einen Plan aus den deterministischen Fakten des Scans. Es führt nie einen Befehl aus, auf keinem Host.

× Ein neuer Versuch, dann eine Ablehnung

Ein ungültiger Plan wird genau einmal neu erzeugt. Der zweite Fehlschlag ist eine Ablehnung, kein Umweg.

× Geschrieben, bevor es läuft

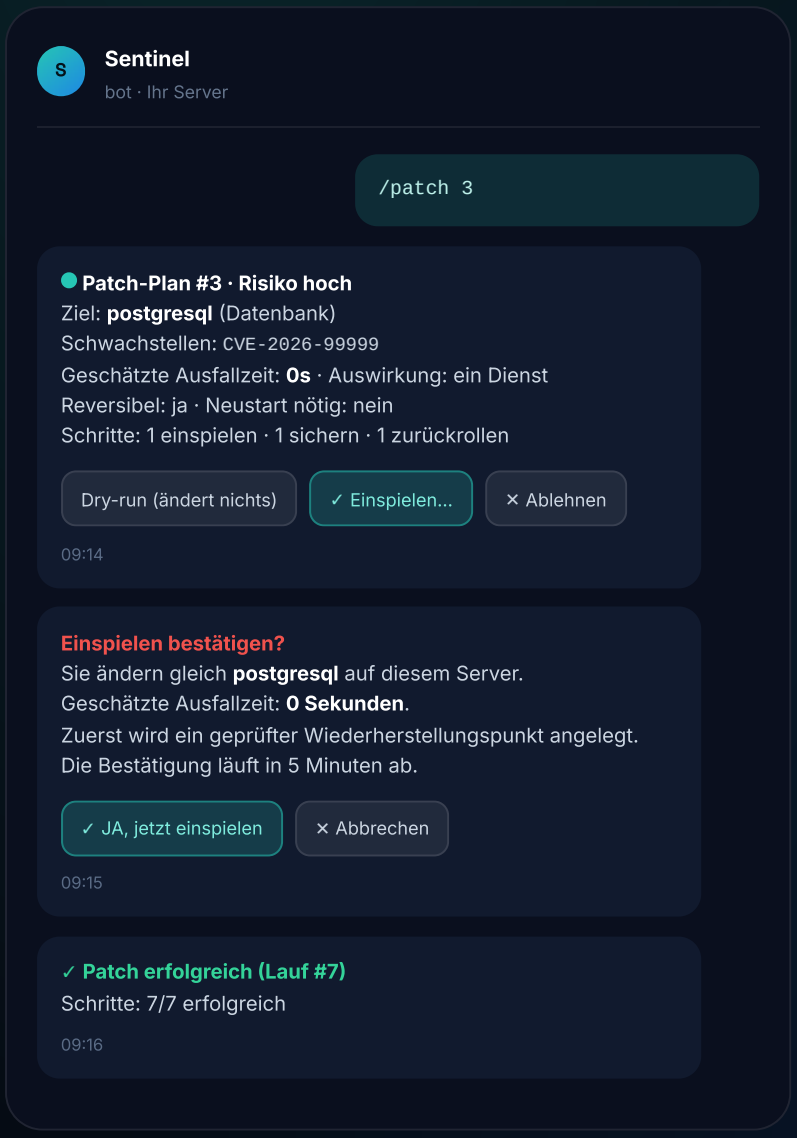
Jeder Schritt landet in der Datenbank, bevor der Befehl ausgeführt wird, damit ein abgebrochener Lauf danach lesbar ist.

Patches werden automatisch erzeugt; eingespielt werden sie nie ohne zwei ausdrückliche Bestätigungen.

PATCH-FREIGABE

Das erste Tippen spielt nichts ein.

Ein Knopf lebt für immer im Chatverlauf. Deshalb *ist* der Knopf nicht die Autorität — er trägt einen Verweis darauf: ein Einmal-Token, an den Chat und an den Plan-Hash gebunden, mit Ablauf.



Der zweite Bildschirm bringt neue Information

Sonst ist „sind Sie sicher?" nur ein Dialog, den man reflexhaft schließt. Er nennt Ziel, Ausfallzeit und was angelegt wird, bevor irgendetwas angefasst wird.

Ein neu erzeugter Plan tötet alte Knöpfe

Wird der Plan zwischenzeitlich neu erzeugt, ändert sich der Hash und jeder frühere Knopf stirbt.

Das Token gilt genau einmal

An den Chat und den Plan-Hash gebunden, mit Ablauf. Wer die Nachricht weiterleitet, leitet nichts Brauchbares weiter.

Das Erzeugen des Plans ist automatisch. Das Einspielen verlangt zwei ausdrückliche Bestätigungen, vom Telefon, von jemandem mit der passenden Rolle.

SICHERHEITSARCHITEKTUR

Ein Agent, der das Internet blockieren kann, wird danach beurteilt, was er verweigert.

Alles Privilegierte läuft durch einen einzigen Root-Prozess von wenigen hundert Zeilen, ohne Abhängigkeiten und ohne Importe aus dem Rest der Anwendung. Der Rest läuft unprivilegiert.

× Es kann Sie nicht aussperren

Zwei unabhängige Schichten: die nftables-Tabelle hat `policy accept` mit einer Allowlist vor den Sperren, und der Executor verweigert die Admin-Adresse rundheraus. Live geprüft — die Verweigerung funktioniert.

× Das Panel kann niemanden sperren

Die Weboberfläche kann eine Sperre nur *aufheben*, nie anlegen. Ein kompromittiertes Panel bleibt ein Datenleck, keine Übernahme.

× `rm` steht nicht auf der Liste

Fehlt in den erlaubten Binaries für Patches. Löschen läuft über eine einzige Operation, strikt auf das Backup-Verzeichnis begrenzt, mit erneut geprüftem Pfad nach der Auflösung.

× Das Wiederherstellungsskript schreibt sich selbst

Erzeugt *innerhalb* des Executors, aus dem, was er auf der Platte sieht. Text von der unprivilegierten Seite wird nie zu einer ausführbaren Zeile, die als root läuft.

× Ein leeres Backup ist kein Backup

Archive werden beim Versiegeln erneut gelesen und gehasht. Ein Artefakt der Größe null wird abgelehnt — sonst sieht es genau so lange wie ein Rückweg aus, bis Sie ihn brauchen.

× Über Logs lässt es sich nicht täuschen

Log-Zeilen werden als nicht vertrauenswürdige Daten verpackt, und das Modell kann nur ein strukturiertes Urteil zurückgeben. Eine gelungene Injektion ändert ein Label, nie das System.

× Die Aufbewahrung löscht den letzten Rückweg nicht

Der letzte Wiederherstellungspunkt eines Assets wird nie gelöscht, egal wie alt. Eine Aufbewahrungsregel, die das kann, ist Datenverlust mit Zeitschaltuhr.

× Not-Leerung

`touch /etc/sentinel/PANIC` leert die Sperrliste alle 10 Sekunden, solange die Datei existiert. Sperren überleben keinen Neustart — Neustarten ist immer ein Ausweg.

INSTALLATION

Ein Befehl, und nichts ändert sich, bis Sie Ja sagen.

Der Wizard erkennt selbst, wo er läuft: auf Ihrem Laptop fragt er nach einem Ziel und installiert über SSH; auf dem Server überspringt er die SSH-Hälfte.

```
$ git clone https://github.com/dragosnimu/sentinel.git
$ cd sentinel && ./scripts/wizard.sh --dry-run
✓ AlmaLinux 9.8 (Familie rhel)
✓ python3.12 — 5208 MB Speicher — 94 GB frei
✓ nginx besitzt 80/443 → Modus „shared“ verfügbar
! Port 8443 ist von etwas anderem belegt
--dry-run: nichts wurde geändert.
```

01 · FRAGT

Nichts wird still geraten

Ziel, Domain, nginx-Modus, Ihre Admin-Adresse, Telegram, der KI-Schlüssel. Wo es einen sinnvollen Standard gibt, sehen Sie ihn, bevor Sie ihn annehmen.

02 · PRÜFT

Nur lesend

Distribution, Speicher, Platte, wer 80/443 besitzt, ob der gewünschte Port frei ist, ob die Domain hierher zeigt. Nichts wird angefasst.

03 · FASST ZUSAMMEN

Der letzte Ausstieg

Alles, was sich gleich ändert, auf dem Bildschirm, vor der Bestätigung. Ein Installer, der nginx schon bearbeitet hat, wenn er Sie fragt, belügt Sie.

04 · INSTALLIERT

Nummeriert und fortsetzbar

Jeder Schritt hat einen Marker auf der Platte. Ist einer Ihrer Dienste stehen geblieben, rollt die Installation sich selbst zurück.

× Secrets landen nie in der Prozesstabelle

Bot-Token und API-Schlüssel werden ohne Echo eingelesen und reisen über stdin. Als Kommandozeilenargumente würde `ps` sie jedem Konto auf dem Host zeigen.

× Zwei Distributionsfamilien, eine Datei

RHEL (AlmaLinux, Rocky, CentOS Stream, Fedora) und Debian (Debian, Ubuntu). Alles, was sich unterscheidet, steht in `deploy/lib/distro.sh`.

× Was nicht unterstützt wird, wird namentlich abgelehnt

`unsupported distribution` — ein Host, auf dem die Installation sichtbar gescheitert ist, lässt sich retten. Einer, der geschützt aussieht und es nicht ist, nicht.

Antworten lassen sich speichern und unbeaufsichtigt wiederholen — ein zweiter Server oder ein Neuaufbau stellt überhaupt keine Fragen.

STAND

Was geliefert ist und in Produktion läuft.

514	31	7	13	0
automatisierte Tests	Sicherheitsinvarianten	systemd-Dienste	Schema-Migrationen	CDN-Abhängigkeiten

Sammlung aus sechs Quellen, deterministische Erkennung, Vorfälle mit KI-Triage, Sperren per Knopf auf Telegram, Schwachstellen-Scan mit KEV-Priorisierung, Erzeugen und Einspielen von Patches mit geprüftem Backup und automatischem Rollback. Eine Weboberfläche ohne fremdes JavaScript, mit strenger Sicherheitsrichtlinie und null externen Ressourcen.

Auto-Block wird **aus** ausgeliefert — die Maschinerie ist vollständig und getestet, aber die ersten 72 Stunden dienen der Beobachtung, damit Sie die Fehlalarme finden, bevor Sie um 3 Uhr nachts jemanden abschneiden. Patches werden automatisch erzeugt; eingespielt werden sie nie ohne zwei ausdrückliche Bestätigungen.

× Was noch nicht abgedeckt ist

Container-Logs werden nicht gesammelt, es gibt über auditd hinaus keinen eigenen Dateiintegritäts-Monitor, und die Debian-Installation ist durch Tests abgedeckt, aber noch nicht auf einem echten Debian-Host verifiziert.

× Wie Sie es bekommen

Installation, Betreut oder Flotte — die Lizenz gilt pro Server, und Ersteinrichtung plus 72-Stunden-Kalibrierung sind in jeder Stufe enthalten.

NÄCHSTER SCHRITT

Dreißig Minuten, kostenlos, unverbindlich.

Sie sagen uns, welchen Server Sie haben und was darauf läuft. Wir sagen Ihnen, was exponiert ist, was Sie gerade angreift und ob Sentinel für Sie Sinn ergibt — auch wenn die Antwort „noch nicht" lautet.

cryptoitdata.eu/contact
ein Gespräch vereinbaren

cryptoitdata.eu/sentinel/de
die Produktseite