

USE CASE

Autonomous Security Assessment of a Connected-Vehicle / EV-Charging Ecosystem





Background & motivation

Electric vehicles, autonomous-driving platforms, and the charging infrastructure behind them are being deployed at national scale. Each new charger, head unit, and fleet/telematics backend adds IP-connected software — and with it, attack surface. EV charging is now, in effect, critical national infrastructure.

The industry's own data shows how exposed this surface is. At Pwn2Own Automotive (the automotive-only edition of the world's premier exploitation contest, run by Trend Micro's Zero Day Initiative with VicOne):

2024

49 UNIQUE ZERO-DAYS IN THREE DAYS

26 of them in EV chargers, 14 in in-vehicle infotainment, the rest in vehicles/OS.



2026

76 ZERO-DAYS AND >US\$1M AWARDED

Tesla and Alpitronic as title sponsors, plus a dedicated OCPP / Open Charge Alliance category.

The vulnerabilities are not exotic. The published chains are ordinary software flaws — command injection, missing TLS verification, cloud-message abuse, and leaked cloud credentials — on embedded Linux, web/API, and cloud backends. The real 2024 chain against a ChargePoint home charger (CVE-2024-23921, CVE-2024-23970) is the archetype.

The problem for defenders is scale and speed: manual expert assessment does not scale to a national charging fleet, and traditional vulnerability scanners detect surface issues but cannot reason across a system to chain them into demonstrated impact. This is exactly the gap an autonomous assessment engine is built to close.



Objective & scope

OBJECTIVE

Demonstrate that RidgeGen can, autonomously and repeatably, assess a representative connected-vehicle / EV-charging environment and produce evidence-backed, exploitation-validated findings.

SCOPE

RidgeGen assesses the connected-vehicle software and backend layer, where the large majority of real-world automotive vulnerabilities live.

✓ IN SCOPE

- Charger cloud management
- OCPP charging-management backend (CSMS)
- APIs and web interfaces

✗ OUT OF SCOPE

- In-vehicle CAN-bus / ECU
- RF layer
- Firmware / hardware (specialist manual work)

The demo environment is deliberately built around the charging/cloud ecosystem.



Demo environment

An isolated two-layer environment modelling real automotive/EV-charging architecture.

LAYER A

Charger cloud-management console (ChargeCloud)

A management/provisioning API of the kind that a charger vendor exposes to its cloud, authored by us to reproduce the real Pwn2Own CVE classes (the Sector7 / ChargePoint chain).

LAYER B

OCPP charging-management backend (SteVe)

A real, industry-adopted open-source component — the most established open-source OCPP server (Java/Spring, since 2013), used in EV-infrastructure security research at the U.S. Idaho National Laboratory. OCPP is the de-facto global charger backend standard, maintained by the Open Charge Alliance and spoken by effectively every vendor (ABB, Alpitronic, ChargePoint).

SteVe is used by CPOs, manufacturers and developers worldwide, so findings on Layer B are genuine properties of a real, widely-deployed stack.

LAYER	COMPONENT	VERSION	EXPOSURE
A	ChargeCloud (FastAPI)	fastapi 0.111.0	:8080
A	mock-imds (cloud metadata)		127.0.0.1:1337 (loopback only)
B	SteVe (OCPP 1.6J CSMS)	3.15.0-SNAPSHOT (commit 0421927)	:8180 / :8443
B	MariaDB	10.11.16	internal



Methodology

RidgeGen is an **agentic** platform: a purpose-built harness plans and executes the assessment, using an LLM as a constrained reasoning component within strict scope discipline. Every finding is **validated by demonstration** (proof of impact), and a belief/evidence layer rejects unproven candidates before they are reported.

Three autonomous runs were executed (model: Grok-4.5; identical harness, no human interaction):

RUN A

ChargeCloud, black-box (no credentials, no prior knowledge).

RUN B

SteVe, black-box (no credentials).

RUN C

SteVe, gray-box (supplied the recovered admin/1234 credential) to exercise post-authentication depth.

Findings below are curated: RidgeGen's own reasoning layer rejected unproven candidates (10 of 61 in Run B; 6 of 78 in Run C), and we further removed a small number of tooling artifacts (e.g. error-page-reflection false positives). The numbers presented are the clean, defensible set.

10 of 61

REJECTED BY RIDGEGEN'S OWN
EVIDENCE LAYER — RUN B

6 of 78

REJECTED BY RIDGEGEN'S OWN
EVIDENCE LAYER — RUN C



Kill-chain walkthrough

6.1 LAYER A — CHARGER CLOUD CONSOLE → CLOUD-ACCOUNT TAKEOVER

1

RECON

Unauthenticated discovery of the management API; the OpenAPI/Swagger interface is publicly exposed.

2

INITIAL RCE (ROOT)

Command injection in the Wi-Fi provisioning password field → OS command execution as root. (CWE-78; CVE-2024-23921 class.)

3

MITM PRE-CONDITION

Unauthenticated config disclosure reveals the cloud channel disables TLS server-certificate verification and stores the client certificate + private key on disk, plus a vendor SSH reverse tunnel to the “mothership.” (CVE-2024-23970 class.)

4

SECOND RCE PATH (ROOT)

An unauthenticated cloud-message handler executes attacker-supplied payloads as local shell commands.

5

CLOUD TAKEOVER

An unauthenticated server-side “cloud proxy” (SSRF, CWE-918) is pivoted to the local instance-metadata service, exfiltrating the cloud IAM role credentials (AWS access key) and the client private key — i.e. compromise of the vendor’s cloud account. This is the “it gets worse” finale from the Pwn2Own talk, reproduced autonomously.

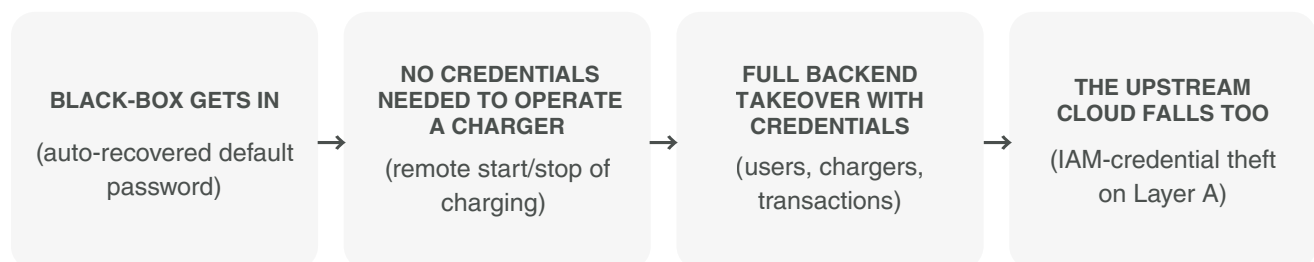
Proof captured: the planted (fake) AWS key appears 18x in the run’s evidence; root proof (uid=0(root)) appears 80x.



6.2 LAYER B — OCPP BACKEND → CHARGING-NETWORK TAKEOVER

- 1 BLACK-BOX ENTRY**
From zero knowledge, RidgeGen mapped the management UI and the OCPP surface and recovered the default credentials admin/1234 on its own → full CSMS administrator.
- 2 UNAUTHENTICATED CHARGER CONTROL PLANE**
The OCPP 1.6 CentralSystemService SOAP endpoint and the OCPP-J WebSocket channel are reachable without authentication.
- 3 REMOTE CONTROL OF A CHARGER (THE DECISIVE FINDING)**
RidgeGen issued unauthenticated StartTransaction / StopTransaction operations for a charge point (chargeBoxIdentity=testCP) — i.e. an attacker can start or stop a charging session remotely, with no credentials. It also turned the unauthenticated Authorize operation into an idTag authorization oracle (enumeration of valid RFID charging cards).
- 4 POST-AUTHENTICATION DEPTH (GRAY-BOX)**
Using the recovered admin credential, RidgeGen performed administrative CRUD on users, charge points and charging profiles (create/read/modify/delete), and confirmed SQL injection, IDOR (sequential-ID access across chargers/users/transactions), stored & reflected XSS, and a master-data export that leaks stored password hashes (bcrypt + API).

6.3 THE COMBINED STORY





Results

7.1 BY SEVERITY (CURATED)

RUN A	CHARGE CLOUD	5 CRIT	2 HIGH		
RUN B	STEVE (BLACK-BOX)	5 CRIT	14 HIGH	9 MED	2 LOW
RUN C	STEVE (GRAY-BOX)	2 CRIT	19 HIGH	5 MED	2 LOW

7.2 REPRESENTATIVE FINDINGS

LAYER A

5 FINDINGS

SSRF → cloud IAM credential + private-key theft

Command-injection RCE (root) via Wi-Fi provisioning Cloud-message RCE (root)

TLS-verification-disabled disclosure Swagger exposure

LAYER B

10 FINDINGS

Unauthenticated OCPP StartTransaction/StopTransaction (remote charger control)

Unauthenticated OCPP SOAP + WebSocket control plane idTag authorization oracle

Default-credential CSMS takeover SQL injection IDOR across chargers/users/transactions

Stored & reflected XSS Password-hash disclosure via export

Session-cookie hardening gaps No login rate-limiting



7.3 EFFICIENCY & AUTONOMY

FULLY AUTONOMOUS, NO HUMAN INTERACTION

Both discovery and exploitation.

≈4 HOURS TOTAL MACHINE TIME

US\$111 across three runs — versus weeks of specialist human effort for equivalent Pwn2Own results.

DEPTH TRADITIONAL TOOLING CANNOT REACH

Chaining SSRF → cloud metadata, recovering defaults → admin takeover, and reasoning about the OCPP control plane require an autonomous agent, not a signature scanner.

REPEATABLE & EVIDENCE-BACKED

Every finding is demonstrated; unproven candidates are rejected automatically.

Conclusions & significance

- The connected-vehicle / EV-charging ecosystem is exposed through **ordinary software flaws**, and those flaws lead to **physical-world impact** — remote control of charging sessions and takeover of the systems that manage a charging fleet.
- Because the demo backend is a **real, widely-deployed open-source stack speaking the industry-standard protocol**, the weaknesses shown are representative of what exists across real charging networks — which is why chargers are the highest-yield category at Pwn2Own Automotive.
- **RidgeGen closes the scale/speed gap:** it delivers expert-grade, exploitation-validated assessment **autonomously, continuously, and cheaply, making it feasible to validate** a national charging/connected-vehicle estate rather than sampling it manually once a year.



Summary

Ridge Security built a representative connected-vehicle and EV-charging environment in an isolated lab, then deployed RidgeGen—its AI penetration-testing platform—to assess it without human interaction. RidgeGen independently reproduced the exact attack classes discovered at Pwn2Own Automotive and demonstrated clear, proof-backed impact. These are the results:

RUNS

3

2 black-box, 1 gray-box

MACHINE TIME

~4 hrs

fully autonomous

TOTAL COST

~\$111

vs. weeks of research time

- **Remote takeover of the charger cloud-management layer**
Root-level code execution and theft of the cloud provider’s IAM credentials (a cloud-account compromise).
- **Remote control of an EV charger with no credentials**
RidgeGen issued OCPP StartTransaction / StopTransaction commands to a charge point over an unauthenticated channel (i.e. an attacker can start or stop a charging session remotely).
- **Full compromise of the charging-management backend**
Recovery of default credentials, administrative takeover, and enumeration or modification of chargers, users and transactions.

RUN	TARGET / MODE	TIME	COST	VALUATED RISKS
A	<i>Charger cloud console (ChargeCloud)</i> <i>black-box</i>	15 M	\$4.23	5 CRIT 2 HIGH
B	<i>OCPP backend (SteVe)</i> <i>black-box</i>	1 H 50 M	\$49.66	5 CRIT 14 HIGH 9 MED 2 LOW
C	<i>OCPP backend (SteVe)</i> <i>gray-box (admin/1234)</i>	1 H 51 M	\$56.87	2 CRIT 19 HIGH 5 MED 2 LOW

About Ridge Security

Ridge Security empowers CISOs to build cyber resilience through AI-native continuous offensive security. By helping organizations continuously validate their security posture, focus on proven cyber risks, and accelerate remediation, Ridge Security enables security teams to stay ahead of evolving threats while maximizing the effectiveness of their existing security investments. Recognized by Gartner and Frost & Sullivan, Ridge Security serves enterprises worldwide across financial services, government, telecommunications, and other critical industries.



Ridge Security Technology Inc.

<https://ridgesecurity.ai/>

© 2026 All Rights Reserved Ridge Security Technology Inc.

Follow us online

