WHITE PAPER

Agentic AI Security | Autonomous Systems

ZipIPS™

Quantum-Secure IoT Authentication with NanoTimestamp Power

Securing Autonomous AI Systems: Authentication for Vehicles, Drones, Robots, and Unmanned Platforms

Helene E. Schmidt, Inventor | Creative Synergies LLC

synergies.com | US10171465B2 | US10348729B2

April 2026

Executive Summary

A self-driving vehicle traveling at highway speed receives a navigation update from an AI traffic management system. An autonomous delivery drone receives a new landing authorization from an AI logistics coordinator. An autonomous warehouse robot receives a routing change from an AI fleet manager. In each case, a physical system moving through the real world is about to execute a command from an AI agent it cannot see, touch, or independently observe.

The authentication question is the same in every case: is this command actually from an authorized source, or has it been spoofed, intercepted, or injected by an adversary? The physical mobility of these systems makes the stakes different from any other ICS environment. A compromised industrial controller affects a facility. A compromised autonomous vehicle affects whatever is in its path.

Legacy authentication answers this question at session initiation and then trusts every subsequent command. ZipIPS™ answers it at the command level — every navigation update, every mission change, every routing instruction is challenged with a fresh timestamp before the autonomous system acts on it. A spoofed or hijacked AI management system cannot produce a valid credential. The command does not execute.

This White Paper examines the authentication challenges specific to civilian autonomous AI systems — self-driving vehicles, delivery drones, autonomous robots, maritime systems, and agricultural automation — and explains how ZipIPS™ addresses them. Grok 4 (xAI) first determined that ZipIPS™ exceeds current NIST post-quantum and lightweight cryptography standards in both security level and credential efficiency. Claude (Anthropic) subsequently reached the same conclusions independently. ZipIPS™ is available for licensing. Creative Synergies LLC welcomes inquiries.

1. The Moving Target Problem

1.1 What Makes Autonomous Systems Different

Industrial control systems are fixed. A compromised PLC at a chemical plant is dangerous, but it is stationary — its consequences are bounded by the physical environment it controls. Autonomous systems are not fixed. They move through shared public spaces, alongside other vehicles, under populated airspace, across agricultural land, through warehouse facilities shared with human workers.

This mobility changes the consequence profile of an authentication failure fundamentally. A fraudulent command to a stationary industrial controller produces consequences within a defined physical boundary. A fraudulent command to an autonomous system in motion produces consequences that are unbounded, unpredictable, and potentially immediate. The physical world does not pause while authentication is reconsidered.

AI-controlled autonomous systems add a further dimension: the commands arrive continuously, at machine speed, from AI management systems that the autonomous platform has no direct way to observe or evaluate. The platform has only the authentication layer to tell it whether the command it just received is genuine.

1.2 The Authentication Gap in Autonomous Systems Today

Current autonomous system authentication relies primarily on encrypted communication channels — TLS, certificate-based PKI, or proprietary protocols — established at connection time. Once the channel is established, commands flowing through it are trusted. This creates three specific vulnerabilities:

- Channel compromise: An attacker who compromises the communication channel between an AI management system and an autonomous platform can inject commands that appear to come from the legitimate management system. The autonomous platform has no mechanism to independently verify the source at the command level.
- Management system hijacking: An attacker who compromises the AI management system itself — or substitutes a fraudulent one — can issue commands to every autonomous platform under its control. All of those commands arrive through authenticated channels and are trusted accordingly.
- Quantum vulnerability: The encryption underlying most autonomous system communication relies on mathematical hardness assumptions that quantum computing threatens. Autonomous systems being deployed today may be operating when quantum computers capable of breaking those assumptions are available to adversaries.

An autonomous vehicle traveling at highway speed cannot pause for a human to verify an incoming navigation command. ZipIPS™ requires no human in the loop — authentication is fully automated, machine-to-machine, with no latency introduced by human review. Whether the implementation meets the real-time response requirements of any specific platform is the licensee's responsibility to evaluate and verify.

1.3 Why On-Demand Authentication Matters

Autonomous systems operate on tight computational and power budgets. A self-driving vehicle's onboard computing resources are managing sensor fusion, path planning, obstacle avoidance, and vehicle control simultaneously. An autonomous drone is managing flight dynamics, navigation, obstacle detection, and payload management on a power-constrained platform. Authentication overhead that consumes continuous background resources competes directly with the primary mission.

ZipIPS™ generates credentials on demand only — no background process, no continuous cryptographic computation, no idle overhead. Authentication resources are consumed only when a command challenge occurs. Between challenges, the autonomous system's computational resources are fully available for navigation, control, and mission execution.

2. How ZipIPS™ Secures Autonomous AI Systems

2.1 Command-Level Challenge Before Execution

The core mechanism is simple. Before an autonomous system executes a command from its AI management system — a navigation update, a mission change, a routing instruction, a parameter adjustment — it generates a fresh timestamp and sends it as a challenge to the commanding source. The commanding source must respond with a credential derived from pre-shared tables that were provisioned before the mission began and never cross the network during operation.

The autonomous system independently derives the expected credential from its own copy of the same tables and compares. If they match, the command executes. If they don't, the command is blocked and permanently logged. The entire exchange happens at machine speed — fast enough for real-time autonomous operation, lightweight enough for power-constrained platforms.

2.2 The Double Two-Way Handshake

Authentication works in both directions. The AI management system verifies the autonomous platform before accepting its sensor reports and status updates. The autonomous platform verifies the AI management system before executing its commands. Neither side trusts a single exchange. Both must pass.

For autonomous systems, the bidirectional property addresses a specific and serious threat: a substituted AI management system that has been inserted between the legitimate management platform and the autonomous fleet. The autonomous platforms challenge the management system before executing each command. A substituted system without the pre-shared tables cannot pass the challenge. The fraudulent command chain is broken.

2.3 Key Properties for Autonomous System Security

Property	Why It Matters for Autonomous Systems
Machine-speed authentication	Autonomous systems cannot pause for slow authentication processes. ZipIPS™ credential generation and verification operates at the speed of the device's internal clock — fast enough for real-time autonomous command processing.
No static credential	Static API keys and session tokens are high-value targets. A compromised credential gives persistent access to every autonomous platform it authenticates. ZipIPS™ generates a fresh credential at every command exchange — there is nothing persistent to steal.
No clock synchronization	Autonomous platforms operating in varied environments may have unreliable access to time synchronization infrastructure. ZipIPS™ requires no synchronized clocks — each platform uses its own internal clock, and the commanding source sends its timestamp as part of the challenge.
On-demand generation only	No background computational overhead. Critical for power-constrained autonomous platforms — drones, sensors, edge devices — where every milliwatt of computing power matters.
Command-level challenge	Every mission-critical command is verified before execution — not just the session. A hijacked or substituted AI management system cannot issue executable commands without the pre-shared tables.

Quantum-secure by construction	Autonomous systems deployed today may still be operating when quantum computing attacks on legacy cryptography become practical. ZipIPS™ is quantum-secure by architecture — no migration required as the threat matures.

3. Autonomous System Applications

3.1 Self-Driving Vehicles

Self-driving vehicles receive continuous commands from AI management systems — navigation updates, traffic management instructions, route changes, speed advisories, and emergency alerts. They also receive software updates, configuration changes, and fleet management instructions from cloud-based AI platforms. Each of these is an authentication event that legacy systems leave unverified at the command level.

The consequences of a fraudulent command to a self-driving vehicle in traffic are immediate and physical. A spoofed navigation update that redirects a vehicle into oncoming traffic, a fraudulent emergency alert that causes a sudden stop on a highway, a hijacked fleet management command that coordinates multiple vehicles into a dangerous configuration — these are not hypothetical scenarios. They are the natural consequence of deploying AI-commanded autonomous vehicles without command-level authentication.

ZipIPS™ authenticates every command before the vehicle acts on it. A navigation update from a spoofed traffic management system cannot produce a valid credential. The vehicle continues on its verified route. A fraudulent emergency alert cannot pass the timestamp challenge. The vehicle responds only to verified commands from its authorized management system.

3.2 Autonomous Delivery Drones

Autonomous delivery drones receive mission instructions, navigation updates, airspace coordination commands, and landing authorizations from AI logistics and air traffic management systems. They operate in shared airspace, over populated areas, carrying payloads that may include valuable or sensitive items. A hijacked drone is an airborne platform that can be redirected, grounded in a dangerous location, or used to interfere with other aircraft.

The FAA's Remote ID requirements for drones — which mandate that unmanned aircraft broadcast identification and location — do not address the authentication of commands the drone receives from its management system. ZipIPS™ addresses exactly that gap: the commands arriving at the drone, not the signals it broadcasts. Every mission update, every airspace routing change, every landing authorization is challenged before execution.

3.3 Autonomous Warehouse and Logistics Robots

Autonomous mobile robots in warehouse and logistics environments receive routing instructions, task assignments, charging schedules, and coordination commands from AI fleet management systems. They operate in facilities shared with human workers, carrying loads, navigating narrow aisles, and interacting with conveyor systems and loading docks. A fraudulent routing command that sends a loaded robot into a human work area, or a hijacked fleet management instruction that coordinates multiple robots into a collision, has immediate safety consequences.

ZipIPS™ authenticates every routing instruction and task assignment before the robot acts on it. A compromised fleet management AI cannot issue executable commands to the robot fleet without the pre-shared tables. The robots continue executing only verified instructions from their authorized management system.

3.4 Autonomous Maritime Systems

Unmanned surface vehicles (USVs) and autonomous underwater vehicles (AUVs) receive navigation, mission, and control commands from remote AI management systems, often over satellite or long-range radio links with significant latency. They operate in open water, in shipping lanes, and in environmentally sensitive areas where a misdirected vessel or a compromised mission profile can have serious consequences.

The intermittent and latency-affected nature of maritime communications makes authentication architecture that requires continuous connectivity impractical. ZipIPS™ requires none — each vessel uses its own internal clock, pre-shared tables are loaded before departure, and authentication operates entirely from local resources regardless of communications conditions. A command that gets through the maritime link also carries its authentication challenge. There is no separate authentication infrastructure to degrade.

3.5 Agricultural Autonomous Systems

Autonomous tractors, sprayers, harvesters, and agricultural drones receive mission instructions, field mapping updates, application rate commands, and coordination instructions from AI farm management systems. They operate on large agricultural properties, often without direct human supervision, carrying equipment and materials — fertilizers, pesticides, herbicides — whose misapplication has significant environmental and crop consequences.

A fraudulent application rate command to an autonomous sprayer — one that doubles the pesticide concentration or redirects application to the wrong field — is not a cybersecurity incident. It is an agricultural incident with real economic and environmental consequences. ZipIPS™ authenticates every application command before execution. A compromised farm management AI cannot direct autonomous agricultural equipment without the pre-shared tables.

4. Threat Scenarios and ZipIPS™ Response

Threat Scenario	Attack Method	ZipIPS™ Response
Spoofed management system	Adversary impersonates the legitimate AI management system, issuing fraudulent navigation or mission commands to autonomous platforms.	Autonomous platform challenges the commanding source with a fresh timestamp. Spoofed system without pre-shared tables cannot respond correctly. Command does not execute.
Man-in-the-Middle command injection	Adversary intercepts the command channel between AI management system and autonomous platform, modifying commands in transit.	Double two-way handshake detects the substitution. The intercepting relay cannot independently produce credentials for both sides of the challenge. Detected and logged.
Replay attack	Adversary captures a valid command credential and attempts to reuse it to issue a fraudulent instruction.	Every credential is tied to a single, unrepeatable and unpredictable timestamp. A captured credential is immediately worthless. There is nothing to replay.
Fleet-wide hijacking	Adversary compromises the AI management system itself, gaining the ability to issue fraudulent commands to every autonomous platform in the fleet.	Each platform independently challenges the management system before executing commands. A compromised management system that has lost its pre-shared tables — through rotation or revocation — cannot issue executable commands. Table rotation is the primary governance control.

Software update injection	Adversary intercepts or spoofs a software update command to an autonomous platform, installing malicious code under the appearance of a legitimate update.	Software update commands authenticated with ZipIPS™ must pass the same timestamp challenge as any other command. A spoofed update command cannot produce a valid credential.

5. Security Performance

ZipIPS™ is a licensable architecture. Implementing organizations configure the system for their specific autonomous platform and operational environment. The following illustrative entropy levels represent what the architecture can achieve. There is no upper limit — entropy can be increased by adding character sets, time units, or any real-time data the implementing system can generate.

Implementation	Entropy	Payload	Characteristics
ms original	381 bits	95 bytes	Millisecond-resolution timestamps. Alphanumeric character strings. Eight time units. Suitable for power-constrained autonomous platforms including delivery drones, agricultural sensors, and lightweight autonomous robots.
ns standard	476 bits	117 bytes	Nanosecond-resolution timestamps. Alphanumeric character strings. Ten time units. Appropriate for self-driving vehicle command authentication, autonomous maritime systems, and warehouse robot fleet management.
ns+ high security	735 bits	157 bytes	Maximum time-unit depth with non-control special characters from the extended ASCII character set. Designed for highest-consequence autonomous systems and sensitive mission command channels.

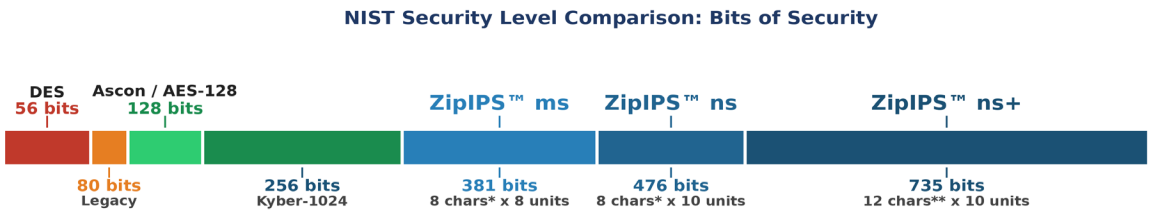

Figure 1: NIST Security Level Comparison — ZipIPS™ illustrative implementations vs. reference points

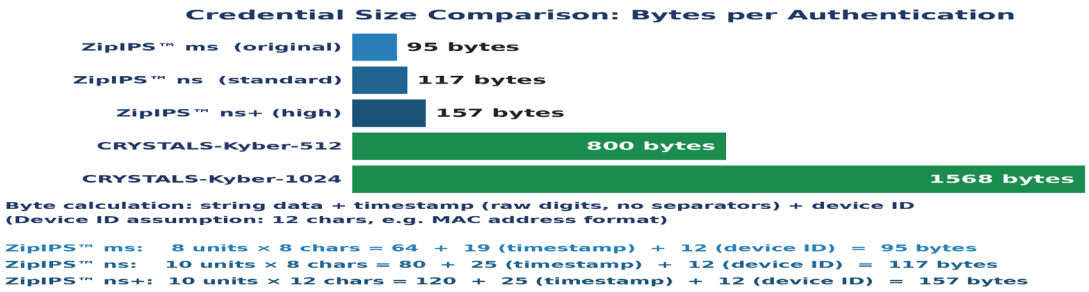

Figure 2: Credential Size Comparison — ZipIPS™ illustrative implementations vs. CRYSTALS-Kyber

A delivery drone authenticating AI commands with ZipIPS™ ms original generates a 381-bit credential in a 95-byte payload — entirely on demand, with no background computational overhead, consuming no power between authentication events. The drone spends its power budget flying and navigating, not running continuous cryptographic processes.

6. Regulatory and Standards Alignment

Standard / Regulation	Requirement	ZipIPS™ Alignment
FAA Remote ID (Drones)	Unmanned aircraft must broadcast identification and location. Command authentication is not addressed by Remote ID but is required for safe autonomous operation.	ZipIPS™ addresses the command authentication gap that Remote ID does not cover — verifying that instructions received by the drone come from an authorized source before execution.
UNECE WP.29 (Autonomous Vehicles)	UN vehicle cybersecurity regulations requiring manufacturers to implement cybersecurity management systems for connected and autonomous vehicles	ZipIPS™ provides command-level authentication addressing the vehicle cybersecurity requirements WP.29 identifies for V2X and remote management command channels.
SP 800-207 (Zero Trust)	Never trust, always verify — no system is implicitly trusted based on prior session or network position	ZipIPS™ implements Zero Trust at the command level for autonomous systems. Every instruction is independently challenged before execution regardless of session state.
FIPS 203/204/205 (PQC Standards)	Transition to quantum-safe cryptography — transportation and autonomous systems infrastructure identified as requiring early transition	ZipIPS™ is quantum-secure by architecture. Autonomous systems authenticated with ZipIPS™ meet the quantum-safe requirement from day one of deployment.
ISO 21434 (Automotive Cybersecurity)	Road vehicle cybersecurity engineering standard, including requirements for authentication of commands received by vehicle systems	ZipIPS™ command-level authentication addresses the vehicle system authentication requirements identified in ISO 21434 for connected and autonomous vehicle architectures.

Note: These alignments are architectural observations, not certifications. ZipIPS™ has not been submitted to the FAA, UNECE, ISO, or any regulatory body for evaluation or endorsement. Prospective licensees should conduct their own evaluation of compliance applicability for their specific autonomous system deployment.

7. Implementation Considerations

ZipIPS™ is a licensable architecture — not a product. Implementing organizations write their own code, configured for their specific autonomous platform, communications architecture, and operational requirements. For autonomous systems specifically, three implementation decisions are important:

- Command-level integration: The most effective implementation builds the timestamp challenge into the platform's command-processing logic — every received command is challenged before execution. For autonomous systems, this means the challenge happens on the platform itself, not at the network edge, ensuring that fraudulent commands are blocked before the platform acts on them regardless of how they entered the communication channel.
- Platform resource calibration: ZipIPS™ on-demand generation is compatible with resource-constrained autonomous platforms. The implementing organization should evaluate the computational and power capabilities of their specific platform and configure the implementation accordingly — ms original for the most constrained platforms, ns standard or ns+ for platforms with greater computational headroom.
- Table provisioning before deployment: Pre-shared tables are loaded during platform provisioning before mission deployment. For autonomous systems that may operate for extended periods without returning to base, the implementing organization should define table rotation procedures appropriate to their operational tempo and accessibility constraints.

Implementation requirements will vary by platform type, communications architecture, and operational environment. These are the licensee's responsibility to evaluate. Creative Synergies LLC makes no representation regarding specific integration requirements for any particular autonomous system deployment.

8. Patent Foundation

ZipIPS™ is protected by two issued United States patents, both assigned to Helene E. Schmidt of Creative Synergies LLC.

Patent	Issued	Scope
US10171465B2	Jan. 1, 2019	Method patent covering the authentication process: timestamp generation, character string retrieval and sequencing, initiating string construction and transmission, host-side verification, second timestamp generation, client-side verification, and the on-demand re-verification loop. Applicable to any networked device pair, including AI management systems commanding autonomous vehicles, drones, robots, and unmanned platforms.
US10348729B2	Jul. 9, 2019	System patent covering the authentication architecture: host device with sequence tables and string tables at variable security levels, client device with mirrored table architecture, device identifier, and the system configuration for the complete double two-way handshake with on-demand re-verification. Continuation of US10171465B2.

9. Licensing

Creative Synergies LLC welcomes inquiries from qualified organizations interested in exploring licensing opportunities. No terms are presented in this White Paper — the right licensing structure is best arrived at through direct dialogue between technically and legally informed parties.

ZipIPS™ is available for licensing. Please visit synergies.com and Contact Us if you have any questions. Helene E. Schmidt, Inventor Creative Synergies LLC synergies.com | zipips@synergies.com

ZipIPS™ is a trademark of Creative Synergies LLC. Protected by U.S. Patents US10171465B2 and US10348729B2. All rights reserved. This document describes technology available for licensing but does not constitute a binding offer or agreement. No licensing terms are expressed or implied. Alignments with regulatory standards are observational and do not represent government or standards body endorsement or certification.