SpoofNet

AI-Assisted Tactical Electromagnetic Deception Research

Public, Non-Confidential Research White Paper

Prepared by	Matthew Daxner; Michael Kurdahi
Affiliation	University of Southern California, M.S. Computer Science
Document scope	Technical research summary of the SpoofNet concept, system architecture, evaluation methodology, and current design findings.
Distribution note	Prepared as a public, non-confidential research summary for open distribution. This paper omits classified, controlled, proprietary sponsor, band-specific transmit, and step-by-step implementation details. RF experimentation is assumed to occur only in simulation, receive-only analysis, laboratory settings, or fully authorized test environments.

Executive Summary

This research paper was an early phase of SpoofNet. It investigates: whether low-cost, distributed edge systems can generate believable electromagnetic behavior that reduces the reliability of adversary ISR capabilities without relying on large, centralized EW platforms? The project studies tactical deception as a software-defined, human-configurable, and AI-assisted systems problem. Instead of treating spoofing as static playback, SpoofNet models deception as coordinated behavior across roles, timing, operator intent, and evaluation metrics.

1. Abstract

Spoofnet is motivated by a widening gap between the pace of ISR-driven targeting and the limited set of deception tools available to small units operating at the tactical edge. Existing electronic warfare capabilities often emphasize specialized platforms, dedicated EW teams, or large-scale effects. SpoofNet examines a different approach: a distributed, low-cost, edge-deployable system that can create believable, scenario-aligned signal activity.

The technical contribution is not only a parts list. It is a system model for translating operator intent into plausible electromagnetic behavior. The system combines an AI scenario engine that creates structured communication plans, a distributed orchestration layer that decomposes those plans into node-level schedules, and an operator-facing interface that enables mission-level control.

This public version presents the research thesis, system rationale, technical architecture, evaluation methodology, current design lessons, and unresolved problems. It intentionally avoids operational transmit procedures, detailed RF configuration, or instructions that would enable unauthorized emissions. The goal is to communicate the engineering and research substance of the project while keeping the document suitable for open distribution.

2. Research Motivation

Modern military operations increasingly depend on the speed of observe, identify, target, and act cycles. In this environment, electromagnetic signatures become tactical evidence. Radio traffic, device emissions, command-post patterns, and other observable signals can reveal unit size, disposition, movement, and operational tempo. As ISR systems and signal intelligence pipelines improve, small units face greater risk from the signatures they create simply by communicating and operating.

The simplest response is emission control: communicate less, transmit less, and reduce observable activity. That can reduce exposure, but it also imposes operational costs by slowing coordination and degrading situational awareness. SpoofNet begins from a different premise: tactical units need the ability not only to hide from sensors, but to shape the inferences those sensors make. Deception becomes useful when it creates ambiguity, consumes adversary attention, or makes real positions harder to distinguish from plausible alternatives.

The research motivation is both practical and technical. If adversary systems infer ground truth from RF behavior, then a deception platform should be evaluated by its ability to produce behavior that is internally coherent, contextually plausible, and difficult to dismiss. This connects generative AI, distributed systems, embedded computing, human-computer interaction, signal realism, and controlled field-test methodology into one engineering problem.

3. Operational Gap and Technical Problem Definition

3.1 Tactical edge gap

Traditional EW systems remain essential, but many are optimized for completely different use cases. There exists a major gap for small ground systems that are capable of localized deception quickly and with limited preparation. A tactical deception platform must be portable, configurable and easily deployable.

SpoofNet frames the gap as a behavior generation and system design problem. A believable deception effect requires consistency across troop disposition, timing, message structure, scheduling, and mission context. The system must answer questions such as which role speaks, how often it communicates, what phase of the mission it appears to be in, how nodes divide responsibility, and how the overall pattern changes over time.

3.2 From repeating signals to coordinated behavior

Basic signal deception can repeat recordings or scripted sequences, but static playback may fail when timing, density, or content does not match the scenario. SpoofNet explores a behavior-oriented model in which a simulated squad, patrol, command element, or support element is represented as a role graph and event timeline. The output is not merely a sequence of messages. It is a coordinated schedule of role-aware actions that can be distributed across multiple nodes and evaluated as a synthetic electromagnetic environment.

3.3 Technical framing

The central technical problem is to transform sparse operator intent into a safe, reviewable, and testable scenario plan. That plan must satisfy multiple constraints: it must be plausible to trained operators, executable on low-cost edge hardware, resilient to node failure or timing drift, and measurable in both simulation and authorized testing. SpoofNet treats these constraints as research variables rather than afterthoughts.

4. Core Thesis and Research Questions

The core thesis of SpoofNet is that constrained AI scenario generation and distributed edge system can create more believable tactical electromagnetic deception than static decoys or single-source playback, while remaining cheaper and more deployable than large EW systems. The thesis rests on four claims:

- **Behavioral realism matters:** Deception effectiveness depends on plausible patterns of unit communication patterns, not only the presence of a waveform or signal.
- **Edge distribution matters:** Multiple coordinated nodes can create spatial and mimic variable troop size that a single source cannot replicate.
- **Human configuration matters:** Operators need mission-level controls and reviewable outputs rather than low-level configuration under time pressure.
- **Evaluation discipline matters:** The project is only credible if believability, observability, usability, robustness, and compliance can be measured against baselines.

The research is organized around the following questions:

Research Question	What the project studies	Why it matters
RQ1: Believability	Can generated traffic patterns appear internally consistent to trained reviewers?	A deception system fails if cadence, role behavior, or message logic is quickly dismissed as implausible.
RQ2: Adaptability	Can behavior vary by mission type, simulated unit size, tempo, and scenario phase?	Static decoys become predictable. Adaptability supports scenario-specific deception research.
RQ3: Distribution	Can coordinated nodes create a more convincing signature than a single source?	Distributed behavior can shape perceived location, size, movement, and command structure.
RQ4: Edge feasibility	Can the approach run on constrained compute, power, thermal, and connectivity budgets?	A tactical edge system must operate under real field-like constraints, not only in cloud simulation.
RQ5: Usability	Can non-specialist operators configure and review scenarios without becoming EW specialized operators?	A useful tool must minimize complexity without abstracting critical information.
RQ6: Responsible testing	Can the system be evaluated through simulation, receive-only analysis, lab tests, and authorized ranges?	Research value depends on repeatable validation without unsafe or unauthorized emissions.

5. Related Work and Prior Validation

The source project materials identify a consistent operational trend: adversaries increasingly use ISR and signal intelligence to accelerate targeting, while small units continue to produce detectable electromagnetic signatures through normal operations. The earlier technical white paper described Army experimentation in which low-cost electronic decoys influenced opposing-force targeting decisions in training environments, supporting the premise that believable low-cost signatures can create tactical ambiguity.

The prior materials also described a broader Army interest in electromagnetic decoy and obfuscation systems, including programs and experiments aimed at raising the noise floor, complicating adversary targeting, and embedding commercial-off-the-shelf innovation directly into field experimentation. SpoofNet is positioned in this research space as a low-cost, software-defined, AI-assisted approach to tactical deception.

The technical opportunity is to move from isolated decoy demonstrations to a repeatable research platform. Many decoys are improvised, scenario-specific, or focused on a narrow signature type. SpoofNet instead asks how deception can be generated, configured, distributed, and evaluated as a system. The objective is to turn tactical ingenuity into a structured platform for studying how operators might create plausible electromagnetic ambiguity.

Stakeholder discovery through defense innovation and academic entrepreneurship channels also shaped the system framing. The strongest feedback was that the highest-value system would not be a standalone transmitter. It would be a configurable deception layer that translates user intent into coordinated behavior across fieldable nodes, while preserving operator review and authorized testing controls.

6. System Architecture

SpoofNet is organized as a layered research architecture. The first layer captures mission context and desired deception effect. The second layer converts that input into a structured scenario, including roles, phases, timing constraints, and candidate message objects. The third layer validates the plan against safety and plausibility constraints. The fourth layer decomposes the approved plan into node-level schedules. The final layer evaluates the output through expert review, receiver-side analysis, simulation, and after-action metrics.

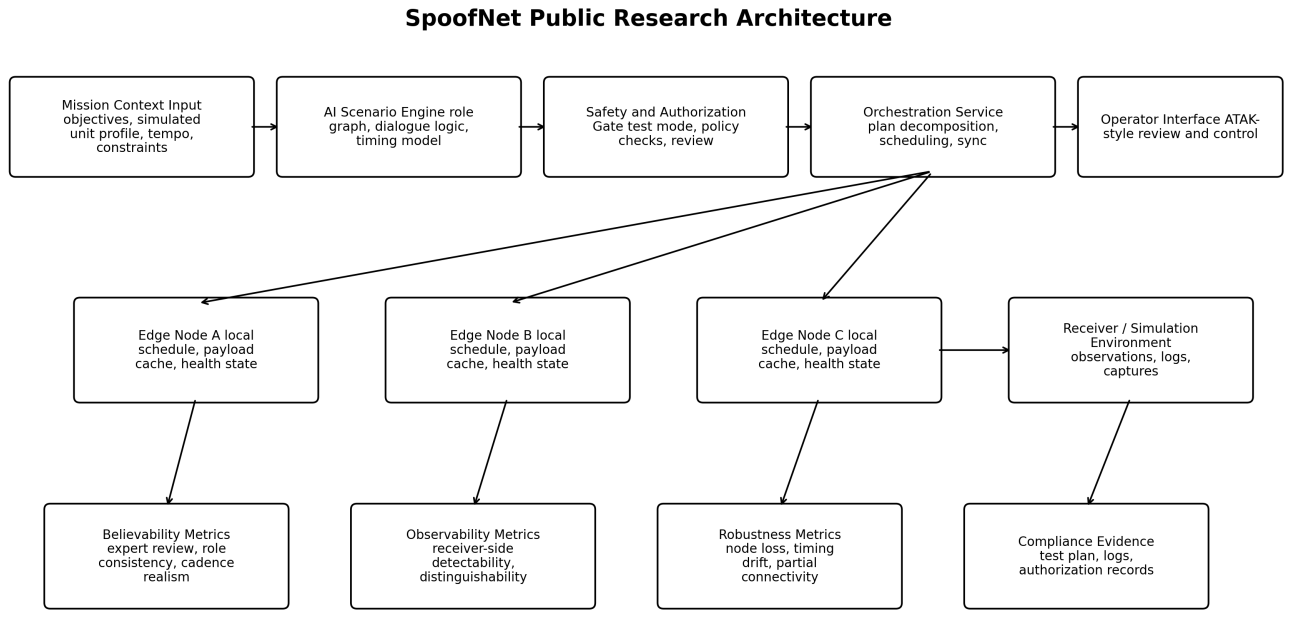

Figure 1. Public research architecture showing the separation between scenario generation, authorization controls, orchestration, edge nodes, and evaluation metrics.

This abstraction separates the research contribution from any single hardware configuration. Hardware will change as power, cost, packaging, and authorization constraints evolve. The research question is broader: how should a tactical deception system represent intent, coordinate behavior, preserve operator control, and measure realism? That separation also makes the platform safer. The same scenario-generation logic can be tested in simulation, receive-only replay analysis, low-power lab setups, or authorized field events without turning the public paper into a deployment guide.

The architecture also separates three planes. The data plane represents the generated behavior under test. The control plane manages configuration, arming, start/stop control, health/status, and schedule delivery. The research plane collects evidence: transcripts, event timelines, receiver observations, reviewer scores, and failure cases. Separating these planes supports modular testing and reduces the risk that a change in one component invalidates the entire research platform.

7. AI-Assisted Scenario Engine

7.1 Structured scenario representation

The AI component is best understood as a constrained scenario generator rather than a free-form text generator. Its input is sparse operator intent: mission type, simulated unit size, desired tempo, approximate duration, role assumptions, and test constraints. Its output is a structured scenario plan that can be reviewed and decomposed. This plan includes role assignments, event phases, message objects, timing windows, node assignments, and metadata needed for evaluation.

Scenario Object	Example Fields	Research Purpose
Mission context	Objective, environment, simulated unit profile, tempo	Defines the scenario boundary and prevents generic traffic generation.
Role graph	Leader, subordinate elements, support roles, communication hierarchy	Improves plausibility by constraining who communicates with whom.
Event timeline	Phase labels, timing windows, expected density, quiet periods	Models cadence and avoids overly regular or unrealistic transmissions.
Message object	Speaker role, intent, abstract channel, text/audio payload reference	Creates reviewable units of generated behavior.
Node assignment	Responsible node, local schedule, fallback behavior, telemetry hook	Turns a global plan into distributed edge execution.
Evaluation metadata	Ground-truth labels, scenario condition, reviewer fields, logs	Allows outputs to be compared against baselines and scored after the test.

7.2 Role-aware communication modeling

A believable simulated unit should not communicate randomly. Leaders communicate differently from subordinate elements. Movement phases differ from coordination phases. A reconnaissance scenario differs from a high-tempo displacement or support scenario. SpoofNet therefore treats role behavior, message cadence, and scenario phase as explicit variables. The goal is controlled variability: enough variation to avoid static signatures, but enough structure to remain internally coherent.

This points toward a hybrid AI architecture. Curated templates and domain-informed rules provide guardrails, while generative models fill in variation, sequencing, and natural-language detail. The AI engine can generate candidate plans, but those plans should pass through validation checks and human review before any emission-capable test mode. In public terms, the important contribution is the constrained planning approach, not a specific model provider or prompt.

7.3 Timing, cadence, and sequencing

Timing is a first-class variable because believability can fail even when individual messages sound plausible. A sequence that is too dense, too regular, too sparse, or inconsistent with the simulated mission phase can appear artificial. SpoofNet can compare deterministic scripts, template-based stochastic schedules, and AI-assisted cadence generation to study which method produces stronger expert-rated realism.

The timing model also creates a systems challenge. A master scenario plan must be split across multiple nodes while preserving global coherence. Each node receives only its relevant schedule, but the overall pattern must still look like one coherent

simulated unit. This requires clock alignment, local schedule caching, and degraded-mode behavior if a node loses connectivity or falls behind.

7.4 Safety-aware generation

The scenario engine should include safety-aware constraints before execution. A plan can be flagged as simulation-only, receive-only, lab-only, or authorized-range-only. Generated output can be separated from any emission-capable path until review and authorization checks are complete. These controls are not administrative extras. They are part of the research design because they allow technical iteration without creating unsafe or unauthorized RF behavior.

8. Distributed Edge Orchestration

The distributed architecture investigates whether multiple low-cost nodes can coordinate to create a stronger deception effect than a single emitter or static playback source. At a high level, each node represents a controllable point in the synthetic electromagnetic environment. The orchestration layer assigns portions of the scenario plan to different nodes so that the resulting behavior appears spatially and temporally distributed.

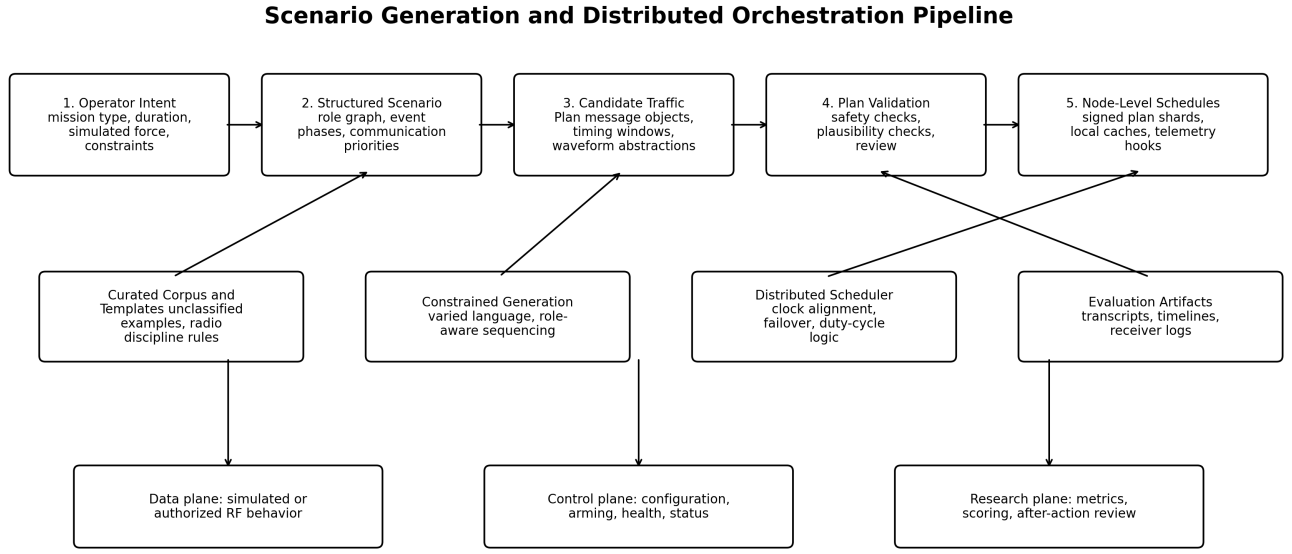

Figure 2. Scenario-to-node pipeline showing how operator intent becomes a structured, validated, and distributed set of node-level schedules.

The source technical materials described a hub-and-node workflow: a group of nodes forms a local subnetwork, one node acts as a hub, the operator configures the scenario through an ATAK-compatible interface, the backend creates a master action plan, and the plan is split into node-specific schedules. This public version preserves the architecture while omitting operational RF parameters. The key research point is the decomposition of a global behavior plan into synchronized local actions.

Subsystem	Technical Role	Design Questions
Hub / controller	Hosts configuration workflow, generates or receives the master scenario plan, distributes schedule shards.	How much logic should remain central versus local? How should review and authorization be enforced?
Edge node agent	Stores local plan, executes simulation or authorized behavior, reports health/status.	What is the minimum local autonomy needed if connectivity changes?
Control link	Provides arming, start/stop, health, and lightweight status messages separate from the behavior under test.	How should control remain resilient without interfering with the primary experiment?
Clock/sync layer	Keeps distributed actions aligned enough for coherent scenario timing.	How much drift is acceptable before believability degrades?
Telemetry/logging	Captures plan IDs, node state, event execution, warnings, and after-action artifacts.	What evidence is needed to reproduce, score, or debug a test?

This is fundamentally a distributed systems problem. The system must handle partial connectivity, timing drift, node drop-out, limited edge compute, local caching, and operator override. A strong implementation should degrade gracefully: if one node fails, the remaining nodes should either continue a reduced plan or pause safely depending on the test mode. That makes robustness a measurable research variable rather than an implementation detail.

9. Edge Prototype Design Considerations

The prior technical materials describe a prototype direction based on commodity edge compute, software-defined radio hardware, out-of-band control, power management, and rugged packaging. In a public research document, the useful technical detail is the set of engineering tradeoffs, not exact transmit configuration. The prototype must balance cost, portability, power draw, thermal headroom, antenna and RF front-end limitations, enclosure durability, and the need for authorized test modes.

A representative node concept contains four functional blocks: edge compute for scheduling and local control, an SDR or simulation adapter for behavior generation under approved conditions, a control/telemetry path for arming and status, and a power/mechanical package suitable for field-like experiments. The specific part choices can change without changing the research architecture.

- **Compute constraint:** Edge nodes must run schedule execution, local status reporting, payload preparation, and fail-safe behavior with limited compute and memory.
- **SDR abstraction:** The public architecture treats waveform behavior as an abstraction layer so simulation, receiver-side analysis, and authorized RF experiments can share the same scenario logic.
- **Control separation:** A lightweight control path such as a low-power telemetry link can be separated from the behavior under test, supporting start/stop, health, and fallback commands.
- **Power and thermal limits:** Battery capacity, amplifier efficiency, duty cycle, and heat dissipation constrain field-like testing and influence node size.
- **Single-radio bottleneck:** A single SDR can generally represent only limited simultaneous behavior, so dense multi-role scenarios require careful scheduling or additional nodes.
- **Mechanical packaging:** Drone placement, carried deployment, and impact resilience create packaging requirements that interact with battery size and antenna placement.

These considerations make the project more technical because they expose the real engineering constraints behind the research. The challenge is not merely to generate text with AI. It is to connect AI-generated scenario structure to edge-executable schedules while respecting power, timing, synchronization, usability, and compliance constraints.

10. Human Factors and Operator Workflow

A deception system designed for tactical use cannot assume that the operator has time to configure low-level technical parameters. SpoofNet therefore emphasizes mission-level configuration. The operator should be able to specify the intended deception effect, simulated unit profile, duration, rough tempo, deployment constraints, and test mode. The system then converts those settings into a coordinated behavior plan that can be reviewed, adjusted, and executed only under authorized conditions.

This creates a human-computer interaction research problem. Too much automation can reduce trust because the operator may not understand what the system will do. Too much manual control slows configuration and increases error. A strong interface exposes the right level of abstraction: enough detail to build confidence and support review, but not so much that the operator must become an EW engineer to use the tool.

The planned ATAK-compatible interface is best described as a research interface for mission-context input and scenario review. Its value is not only that it appears inside a familiar tactical application. Its value is that it embeds deception planning into the same map, route, team, and timing context where operators already reason about missions. This allows operator intent to become structured input for the AI scenario engine and orchestration layer.

11. Evaluation Methodology

A research-oriented paper needs a method for deciding whether the idea works. SpoofNet should be evaluated across believability, observability, usability, robustness, edge feasibility, and compliance. Each category can be studied in increasing levels of realism, beginning with simulation and expert review before moving to controlled laboratory tests or authorized field exercises.

Evaluation Area	Representative Metric	Evidence Collected
Believability	Expert realism score; role consistency; timing plausibility; scenario coherence	Blind review by trained personnel, scoring rubrics, comparison to baseline scripts
Observability	Receiver-side detectability and distinguishability under safe test conditions	Receiver logs, spectrum captures from authorized tests, simulation outputs
Usability	Time to configure scenario; operator error rate; confidence and review quality	Mock missions, interface walkthroughs, after-action interviews
Robustness	Performance under node loss, timing drift, partial connectivity, or local cache failure	Fault-injection tests, schedule recovery analysis, degraded-mode trials
Edge feasibility	Power draw, thermal behavior, local compute headroom, schedule latency	Bench tests, telemetry, resource logs, hardware-in-the-loop tests
Safety and compliance	Adherence to authorization boundaries, test mode, logs, and review gates	Test plans, checklists, approval artifacts, audit logs

The evaluation should compare SpoofNet against simpler baselines. A useful design includes at least three conditions: static playback, template-based scenario generation, and AI-assisted scenario generation. Reviewers would score outputs without knowing which condition produced them. This allows the project to test whether AI assistance improves realism rather than assuming it does.

The evaluation should also separate content realism from system realism. Content realism asks whether messages, roles, and sequence logic appear plausible. System realism asks whether the distributed behavior appears coherent when observed as an environment. Both matter. A transcript can be plausible while timing is unrealistic, and a distributed signal pattern can appear coherent while the underlying scenario makes little operational sense.

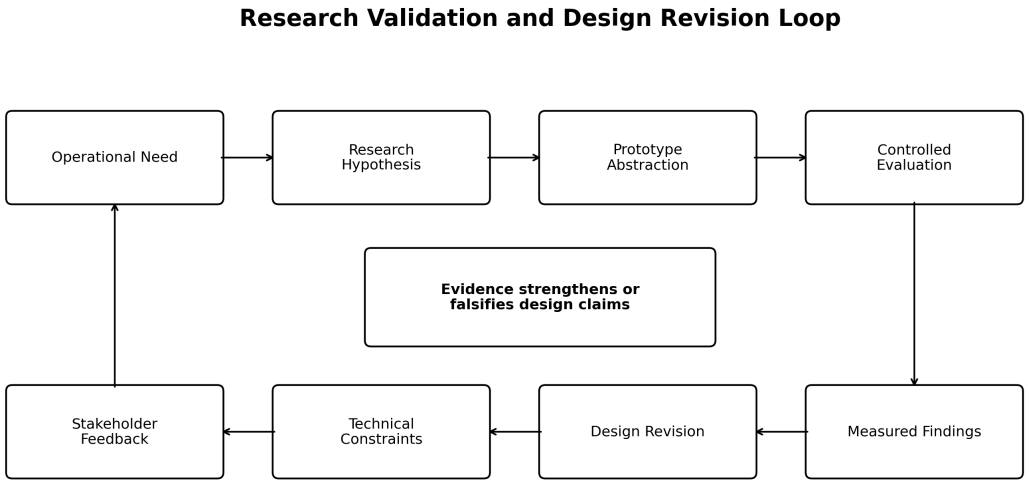

Figure 3. Research loop used to move from operational need to defensible design claims through controlled evaluation and revision.

12. Current Research Findings and Design Lessons

The current state of the project supports several preliminary design lessons. These are not claims of operational effectiveness. They are findings about what the system must model, constrain, and evaluate to become a credible research platform.

- **Finding 1 - Deception requires structure:** Believable deception depends on role hierarchy, message cadence, scenario phases, and timing patterns. A transmitter-only framing is too narrow.

- **Finding 2 - AI is most useful when constrained:** Generative models are valuable for variation and scenario adaptation, but outputs need templates, domain rules, review paths, and safety gates.
- **Finding 3 - Distributed coordination is the core systems challenge:** The hard problem is coordinating many actions across nodes while preserving timing, resilience, operator control, and auditability.
- **Finding 4 - Evaluation access is a limiting factor:** Realistic validation depends on trained reviewers and authorized environments, making simulation and structured expert review essential intermediate steps.
- **Finding 5 - Edge constraints shape the architecture:** Power, thermal behavior, single-radio scheduling limits, connectivity, and packaging constraints directly influence what scenarios are feasible.
- **Finding 6 - Public presentation benefits from abstraction:** A non-confidential paper can still be technically serious when it focuses on architecture, research variables, metrics, and design tradeoffs rather than operational instructions.

Together, these findings solidify the idea. SpoofNet is not simply a proposal to build a device. It is an investigation into how to design, constrain, evaluate, and operationalize AI-generated deception as a system-level capability under responsible research controls.

13. Technical Risks and Open Research Problems

Several unresolved technical problems remain central to the research. First, believability is difficult to quantify. A system may produce outputs that seem plausible to engineers but fail under expert review. The project therefore needs structured scoring by people who understand radio discipline, tactical communication patterns, and ISR analysis.

Second, data scarcity limits model development. Authentic military communication behavior is sensitive and often unavailable. SpoofNet must rely on synthetic corpora, unclassified examples, stakeholder feedback, and carefully designed templates. A key research question is whether synthetic but rule-constrained behavior can be believable enough for controlled training and deception research.

Third, field validation is constrained by authorization and access. The most valuable tests require approved environments, specialized receivers, clear boundaries, and repeatable instrumentation. Without those conditions, the research must rely on simulation, bench testing, recorded outputs, and expert review. Stating this limitation clearly demonstrates technical maturity rather than weakness.

Fourth, edge execution introduces practical risk. Nodes may have limited power, compute, thermal headroom, and connectivity. A robust system needs graceful degradation, local autonomy, simple health reporting, and safe stop behavior. These constraints create meaningful engineering problems that distinguish the project from a software-only AI demonstration.

Fifth, waveform abstraction is an open design challenge. A public research system should avoid exposing sensitive or operational details, but the architecture still needs a way to represent waveform families, timing behavior, channel assumptions, and receiver observations in a way that supports meaningful evaluation. This requires careful abstraction so that simulation and authorized testing remain comparable.

15. Conclusion

This white paper represents the first research phase of SpoofNet: the transition from an initial tactical concept into a more defensible technical framework. The core finding from this phase is that AI-assisted electromagnetic deception is most credible when it is constrained by mission context, role logic, timing models, distributed orchestration, human review, and responsible testing controls. The project's contribution is therefore not limited to hardware integration or signal generation. It is a research-driven validation of how small units might shape the electromagnetic environment using low-cost, configurable, human-in-the-loop systems in authorized settings.

This phase solidified the central design thesis behind SpoofNet. Effective tactical deception cannot be treated as simple signal playback; it requires coordinated behavior that appears plausible across content, timing, spatial distribution, and operator intent. By reframing the project around research questions, system architecture, evaluation criteria, and technical risk, this paper establishes a foundation for future prototype development and controlled validation.

References

1. Joint Publication 3-13.1, Electronic Warfare, U.S. Department of Defense.
2. DefenseScoop, "Cheap electronic decoys providing Army units more flexibility on the battlefield," September 2024, cited in the source white paper.
3. DefenseScoop, "Army electromagnetic decoy and obfuscation systems ramping up for FY25," March 2024, cited in the source white paper.
4. DefenseScoop, "Army transforming in contact approach to fielding innovation," May 2024, cited in the source white paper.
5. SpoofNet source technical white paper, "AI-Driven Tactical Deception for Operational Advantage," provided as project background.
6. USC Hacking for Defense and defense stakeholder discovery notes, summarized in the source materials.