Semantic Remote Control (SRC)

Hardware-Level AI Assistance for Resilient System Administration

Author: **Ali Ganji** (ID: Z23599848) • Supervisor: **Dr. Sayed Pouria Talebi** • School of Computing & Cybersecurity • University of Roehampton

PROBLEM STATEMENT

Modern IT administration relies on software-level management agents (SSH, RDP, cloud APIs), creating a fatal vulnerability: when network settings fail, kernel panics occur, or pre-boot environments (BIOS/UEFI) require intervention, in-band access collapses. Autonomous software agents lose the very channels needed to diagnose and repair failures. Operating system dependency leaves bare-metal and air-gapped systems unreachable without manual, on-site physical intervention, driving server downtime and administrative bottlenecks.

AIMS AND OBJECTIVES

AIMS (WHAT)

AIM 1 (Resilience): Design and construct an out-of-band (OOB) hardware-level AI control bridge using PiKVM and MCP for OS-independent administration.

AIM 2 (Closed Loop): Construct a robust observation-action loop combining HDMI capture, OCR parsing, and USB HID injection for bare-metal states.

AIM 3 (Security): Establish a defense-in-depth governance framework with credential isolation, human-in-the-loop gating, and audit logging.

OBJECTIVES (HOW)

- **OBJ 1** **Limitation Analysis:** Investigate software AI control failure modes in LiLiX/LiLiW prototypes.
- **OBJ 2** **Hardware Build:** Assemble PiKVM v2 node with HDMI-to-CSI video loop and USB HID emulation.
- **OBJ 3** **MCP Tools:** Expose hardware screen capture and keystroke actions via Model Context Protocol.
- **OBJ 4** **Credential Isolation:** Build local security bridge to hide host/PiKVM credentials from AI agent.
- **OBJ 5** **VLM OCR Pipeline:** Deploy vision-language reasoning for screen interpretation and error diagnostics.
- **OBJ 6** **Safety Safeguards:** Implement rate limiting, human approval for power cycles, and cryptographic logs.
- **OBJ 7** **Evaluation Plan:** Structure scenario-based evaluation protocols for BIOS, network, and installer recovery.

ARCHITECTURAL RESILIENCE MATRIX (BY DESIGN)

Scenario / Feature	In-Band (SSH/API)	Blind USB	SRC (Out-of-Band)
BIOS / UEFI Setup	Fails (No OS)	Blind (No feedback)	Full OOB Control
Network Down	Connection Lost	No Verification	Isolated Video Path
Kernel Panic / BSOD	Daemon Dead	Ignored	OCR Screen Diagnostic
Visual Verification	None	None	Real-time HDMI Capture

INTRODUCTION

Autonomous system administration typically presumes operational operating systems and network stacks. However, critical infrastructure emergencies—such as broken routing, crashed kernels, corrupted bootloaders, and air-gapped deployments—sever in-band software channels. Semantic Remote Control (SRC) addresses this paradigm gap by transferring AI interaction to the physical hardware interface.

Evolving from early software-agent prototypes (LiLiX and LiLiW), SRC couples a dedicated PiKVM out-of-band hardware node with a local security bridge and an MCP tool server. This establishes a closed observation-action loop: visual feed acquisition via HDMI capture, vision-language reasoning, and precise hardware keystroke/mouse injection.

Engineered for systems engineers, site reliability teams, and cybersecurity administrators, SRC delivers resilient, auditable autonomous recovery for bare-metal machines, ensuring business continuity during total software isolation without compromising system integrity.

SRC Physical Hardware Node • Assembled Prototype Build

Hardware Central Feature



Studio-Lit Custom Enclosure
Matte-white custom hardware node chassis optimized for passive thermal dissipation and isolated out-of-band rack deployment.

HDMI-to-CSI Video Ingestion
Dedicated TC358743 bridge capturing uncompressed 1080p bare-metal screen frames directly from host GPU without OS drivers.

Isolated USB HID & Power Control
Backpower-protected USB-C OTG interface emulating virtual keyboard/mouse and relay-gated ATX power control.

SYSTEM ARCHITECTURE & TRUST BOUNDARY

AI Client — Claude / LLM Agent, ReAct Tool Calling → **MCP Server** — Standard Protocol, Tool Schema Expose → **Security Bridge** — Credential Proxy, Rate Limit & Auth → **PiKVM v2 Node** — HDMI-to-CSI Loop, USB HID Emulation → **Target Server** — Bare-Metal System, BIOS / Recovery

CLOSED-LOOP PERCEPTION-ACTION FEEDBACK LOOP

1. **HDMI Capture** — 1080p frame grab via CSI-2 bridge
2. **Vision OCR** — Extracts text, errors & cursor state
3. **AI Reasoning** — ReAct agent decides next action
4. **Policy Gate** — Bridge validates rate limits & human approval
5. **HID Injection** — Virtual USB keystroke or mouse event
6. **Verification** — Immediate frame re-capture loopback

PROJECT PLAN & KEY MILESTONES

Project Workstream	Nov 26	Dec 26	Jan 27	Feb 27	Mar 27	Apr 27
Phase 1: Architecture Review & Scope	Done	Done				
Phase 2: Hardware Build & MCP Bridge		Active	Active			
Phase 3: AI Loop & Safety Guardrails			Plan	Plan		
Phase 4: Controlled Trials & Defense				Trials	Poster	Final

Nov 2026 — Architecture Validation & LiLiX/LiLiW review.

Jan 2027 — PiKVM node build, HDMI-CSI & MCP tool server.

Feb 2027 — VLM screen parsing, rate limiting & human gating.

Mar 2027 — Supervised scenario trials & dissertation defense.

PLANNED EVALUATION DESIGN

In accordance with project preparation status, empirical validation will be executed in supervised lab trials across defined administrative scenarios:

1. Network Disruption
Validate AI recovery path when in-band SSH / network stacks drop.

2. BIOS / Boot Navigation
Assess pre-OS visual recognition and keyboard navigation.

3. Recovery Console
Test text entry and diagnostic command parsing during OS crash.

4. Operator Takeover
Verify emergency human interrupt, rate limiting, and power cutoffs.

EVALUATION METRICS TO BE MEASURED:

Task Success Rate (%) · Observation Latency (ms) · Human Intervention Count · Action Error Rate · Recovery Duration · Safety / Policy Violations

LEGAL, ETHICAL AND PROFESSIONAL

SRC operates as an administrative resilience artifact under strict BCS and IEEE Codes of Conduct. Operating at hardware level introduces potential risks of unauthorized access and data exfiltration. Consequently, SRC enforces a strict defense-in-depth posture: target systems are strictly owned or authorized lab hardware; live production deployments and offensive payload generation are out of scope.

To comply with UK GDPR and data privacy standards, captured screen frames are processed in volatile memory, sanitized against PII or credential leakage, and excluded from persistent external model training. Raw PiKVM management credentials remain strictly isolated within a local security bridge, hidden from the AI agent context. High-risk power cycling requires mandatory human-in-the-loop authorization, paired with cryptographically verifiable audit logging to ensure transparency, safety, and regulatory accountability.

USE OF AI STATEMENT

AI is utilized in SRC via multimodal LLMs that interpret HDMI screen captures, extract text through OCR, and synthesize structured MCP tool calls for keyboard and mouse actions. To prevent catastrophic hallucination or drift, the agent operates in an iterative ReAct observation loop: every hardware action must be verified by subsequent frame capture before progressing. High-risk actions (e.g., hard power cutoffs) require explicit human approval. Model prompts are defensively constrained using system guardrails, sandboxed execution, and comprehensive request logging to eliminate prompt injection risks and maintain operational safety.

KEY LITERATURE / REFERENCES (IEEE)

[1] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, Sep. 1975.

[2] S. Yao *et al.*, "ReAct: Synergizing reasoning and acting in language models," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2023, pp. 1–33.

[3] T. Schick *et al.*, "Toolformer: Language models can teach themselves to use tools," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 36, pp. 6853–6871, 2023.

[4] Model Context Protocol, "Model Context Protocol Specification," 2026. [Online]. Available: `modelcontextprotocol.io`

[5] PiKVM Project, "PiKVM: Open and Inexpensive DIY KVM-over-IP," 2026. [Online]. Available: `docs.pikvm.org`