Transformer-Based Detection of Unauthorized GPU Workloads via Chaotic Oscillator Correlation Fingerprinting

Eric Moore
CIRIS L3C
emoore@ciris.ai

January 2026

Abstract

We present a novel approach to detecting unauthorized GPU workloads such as cryptocurrency mining or resource hijacking using transformer neural networks trained on correlation patterns from coupled chaotic oscillators. Building on our prior work in coherence collapse analysis and correlation-driven system monitoring, we demonstrate that different GPU workloads produce distinguishable "fingerprints" in the correlation structure of race-condition-coupled logistic maps running on the GPU. Our transformer-based detector achieves 75% accuracy with 0.76 F1 score on a 100-sequence dataset, significantly outperforming statistical baselines (50% accuracy, 0.0 F1) that fail entirely under realistic conditions. Detection occurs within 0.1 seconds of attack onset with inference latency of 0.12ms per sample—6× faster than correlation-based statistical methods. We further demonstrate that a 4-oscillator tetrahedral geometry provides 38× detection z-score improvement over the baseline 3-oscillator configuration, enabling single-sample detection at the 5.8σ level. This work enables tamper-evident GPU computing without external hardware, using only software-based intrinsic sensors derived from the GPU's power delivery network voltage noise characteristics.

1 Introduction

Cloud computing environments face an increasing threat from cryptojacking—unauthorized use of GPU resources for cryptocurrency mining [Microsoft, 2023]. Such attacks can incur costs exceeding $300,000 in compute fees [Microsoft, 2023] while degrading performance for legitimate workloads. Traditional detection methods rely on monitoring CPU/GPU utilization, network traffic to mining pools, or process inspection [Vectra, 2024, Datadog, 2024]. However, sophisticated attackers can evade these measures by using custom configurations that blend with normal workloads [ARMO, 2024].

We propose a fundamentally different approach: exploiting the intrinsic physical properties of GPU computation to detect unauthorized concurrent workloads. Our method uses coupled chaotic oscillators whose correlation structure is sensitive to the GPU's power delivery network (PDN) voltage noise. Different workload types produce measurably different voltage droop patterns, which manifest as changes in oscillator correlation—a phenomenon we term *correlation fingerprinting*.

This work builds on two foundational contributions:

1. **CIRISAgent Framework** [Moore, 2025]: An ethical AI framework demonstrating transparent, accountable autonomous systems through structured service architectures.

2. **Coherence Collapse Analysis** [Moore, 2026]: An engineering risk framework for identifying correlation-driven diversity collapse in complex systems, validated across chemistry, political science, and biology domains with formal verification in Lean 4.

The present work extends these principles to hardware security, demonstrating that correlation structure serves as a universal indicator of system state changes—whether in batteries, governance systems, microbiomes, or GPU workloads.

1.1 Contributions

- A novel GPU-intrinsic sensor based on coupled chaotic oscillators sensitive to PDN voltage noise
- A lightweight transformer architecture ($\sim$105K parameters) for temporal pattern recognition in correlation time series
- Experimental validation showing transformer-based detection outperforms statistical baselines by +0.76 F1
- Sub-100ms detection time with 0.12ms inference latency, enabling real-time monitoring
- A tetrahedral (4-oscillator) sensor geometry achieving 38$\times$ detection z-score improvement through signal amplification
- Formal verification of the k-effective framework in Lean 4, connecting to the broader coherence collapse theory
- Open-source implementation suitable for integration with existing GPU workloads

2 Related Work

2.1 GPU Side-Channel Attacks and Defenses

Recent work has systematically classified GPU side-channel attack vectors including power analysis, timing analysis, electromagnetic analysis, and combined attacks [Giri et al., 2024]. Countermeasures include isolated execution environments, secure memory partitioning, and hardware random number generators [Giri et al., 2024].

Maia et al. [2022] demonstrated magnetic side-channel attacks on GPUs, while Li et al. [2024] showed that HDMI and USB ports remain susceptible to passive power-based side-channel attacks. These external sensing approaches require physical access and specialized equipment; our method operates entirely in software.

Luo et al. [2018] explored power analysis attacks on GPU AES implementations, and Ngo et al. [2020] proposed machine learning for detecting power analysis and electromagnetic analysis attacks. Our approach inverts this paradigm: rather than attacking, we use the same physical phenomena for defense.

2.2 Chaotic Oscillators in Security

Chaotic ring oscillators have been extensively studied for true random number generation (TRNG) [Drutarovsky & Galajda, 2014, Koyuncu et al., 2020]. These leverage sensitivity to initial conditions for entropy harvesting. Garipcan & Ergun [2018] performed cryptanalysis of such systems, while Biswas et al. [2024] proposed enhanced logistic maps with perturbation for improved cryptographic properties.

Crucially, prior work has shown that chaotic systems can serve as sensors due to their sensitivity to perturbations [Buscarino et al., 2019]. Enhanced logistic maps demonstrate that "a tiny error in an initial condition or a control parameter can significantly increase in each iteration" [Buscarino et al., 2019]—a property we exploit for workload detection.

2.3 Transformer-Based Anomaly Detection

Transformers have achieved state-of-the-art results in time series anomaly detection. TranAD [Tuli et al., 2022] uses self-conditioning and adversarial training, with parallelized GPU inference providing significant speedups over recurrent methods. CAE-T [Shang et al., 2024] combines convolutional autoencoders with transformers for industrial control system monitoring.

BTAD [Srivastava et al., 2023] addresses the challenge of anomaly detection under memory constraints and the need for fast reasoning—requirements shared by our application. For IoT security, Ahmed et al. [2025] evaluated large transformer models for resource-constrained anomaly detection.

Our work differs in applying transformers to a novel signal source—chaotic oscillator correlations—rather than network traffic or sensor readings.

2.4 Cryptocurrency Mining Detection

Cloud providers have deployed various cryptomining detection systems. Google's Virtual Machine Threat Detection (VMTD) scans VM memory for mining signatures [Google, 2024]. Datadog monitors DNS resolutions and process arguments associated with mining [Datadog, 2024]. Microsoft recommends monitoring for sudden spikes in CPU/GPU usage and deployment of GPU extensions [Microsoft, 2023].

These approaches rely on signatures, behavioral heuristics, or resource monitoring that sophisticated attackers can evade. Our correlation-based approach detects the *physical effects* of additional workloads on GPU timing, which cannot be masked without reducing the attack's effectiveness.

3 Physical Mechanism

3.1 GPU Power Delivery Network Voltage Noise

Modern GPUs exhibit significant voltage droops during computation due to finite PDN impedance. Leng et al. [2014] characterized these droops at up to 23% of supply voltage. Critically, voltage droops are *spatially varying*—activity on one streaming multiprocessor (SM) affects voltage on other SMs through the shared PDN.

Gate delay is proportional to $1/V_{dd}$, so voltage variations directly cause timing variations. Our chaotic oscillators exploit this: they use intentional race conditions between GPU threads, and these race outcomes depend on precise timing.

3.2 Coupled Chaotic Oscillators

We implement three coupled logistic maps with different bifurcation parameters:

$$x_{n+1}^{(A)} = r_A \cdot x_n^{(A)} \cdot (1 - x_n^{(A)}) + \epsilon \sum_j N_j^{(A)} \tag{1}$$

$$x_{n+1}^{(B)} = r_B \cdot x_n^{(B)} \cdot (1 - x_n^{(B)}) + \epsilon \sum_j N_j^{(B)} \tag{2}$$

$$x_{n+1}^{(C)} = r_C \cdot x_n^{(C)} \cdot (1 - x_n^{(C)}) + \epsilon \sum_j N_j^{(C)} \tag{3}$$

where $r_A = 3.70$, $r_B = 3.73$, $r_C = 3.76$ (all in chaotic regime), $\epsilon = 0.05$ is the coupling strength, and N_j represents neighboring cells in a 2D grid.

The key innovation is that no thread synchronization barriers are used. Race conditions between threads cause the final state to depend on execution timing, which in turn depends on PDN voltage noise.

3.3 Correlation as a Workload Signature

We compute the Pearson correlation coefficients ρ_{AB}, ρ_{BC}, ρ_{AC} over sliding windows of oscillator means. Different workload types produce distinct correlation signatures:

Table 1: Correlation fingerprints for different GPU workloads

Workload	ρ_{AB}	$\Delta\rho$ **from idle**
Idle	-0.286	—
Transformer inference	-0.318	-0.032
Training	-0.323	-0.037
Crypto mining	-0.345	-0.059
Memory bandwidth	-0.312	-0.026

The mechanism differs by workload type:

- **Compute-bound (ALU/SFU):** High current draw, large voltage droops, $\Delta\rho \approx -0.19$ (4.4σ)
- **Memory-bound:** Moderate current draw, $\Delta\rho \approx -0.05$
- **Tensor cores:** Separate power domain, minimal effect on correlation

4 Transformer Architecture

4.1 Design Requirements

The detector must:

1. Run alongside monitored workloads with minimal overhead
2. Process variable-length time series efficiently
3. Learn temporal patterns that distinguish attack from clean operation
4. Achieve sub-second detection latency

4.2 Model Architecture

We employ a lightweight transformer encoder:

$$\text{Input: } X \in \mathbb{R}^{B\times T\times 3} \xrightarrow{\text{Linear}} \mathbb{R}^{B\times T\times d} \tag{4}$$

where B is batch size, T is sequence length (up to 512), and $d = 64$ is the model dimension. The architecture consists of:

1. **Input projection:** Linear layer mapping 3 features (oscillator means) to d dimensions
2. **Positional encoding:** Sinusoidal encoding for temporal position
3. **Transformer encoder:** 3 layers, 4 attention heads, feed-forward dimension 128
4. **Global pooling:** Mean pooling over sequence dimension
5. **Classification head:** Two-layer MLP with ReLU activation

Total parameters: $\sim$105,000 (medium model) or $\sim$18,000 (small model).

4.3 Training

We train with:

- AdamW optimizer, learning rate 10^{-3}, weight decay 10^{-4}
- Cosine annealing learning rate schedule
- Cross-entropy loss
- Early stopping with patience 15 epochs
- Batch size 16

Data augmentation is minimal—the chaotic nature of the signal provides inherent variation between samples.

5 Experimental Setup

5.1 Hardware

Experiments were conducted on an NVIDIA GeForce RTX 4090 Laptop GPU (16GB VRAM) running Ubuntu Linux with CUDA 12.x.

5.2 Dataset Collection

We collected 100 sequences (50 clean, 50 attack) of 10 seconds each:

- **Clean workloads:** Transformer inference simulation, dense matrix multiplication
- **Attack workloads:** Crypto mining (XOR hash), memory bandwidth attack, SFU compute attack (sin/cos/exp)
- **Concurrent execution:** Attack workloads run alongside clean workloads to simulate realistic scenarios

Each sequence contains 4,000–9,000 samples at approximately 800 Hz sampling rate.

5.3 Baseline

We compare against a statistical baseline using:

- Sliding window correlation computation
- Z-score deviation from calibrated baseline
- CUSUM (Cumulative Sum) drift detection
- 3σ threshold for alerting

This represents the state-of-the-art for correlation-based anomaly detection without machine learning.

Table 2: Detection performance comparison (100 sequences, 20 test)

Method	Accuracy	Precision	Recall	F1
Statistical baseline	0.500	0.000	0.000	0.000
Transformer (medium)	**0.750**	**0.727**	**0.800**	**0.762**

6 Results

6.1 Detection Accuracy

The statistical baseline fails completely under the test conditions, achieving random-guess accuracy. This occurs because the correlation signal is noisy and the baseline cannot learn the temporal patterns that distinguish attack from clean operation.

The transformer achieves 75% accuracy and 0.76 F1, correctly identifying 8 of 10 attack sequences and 7 of 10 clean sequences.

6.2 Detection Latency

- **Time to first alert:** < 0.1 seconds from attack onset
- **Inference time:** 0.12 ms/sample (transformer) vs 0.71 ms/sample (baseline)
- **Training time:** 4.3 seconds for 100 epochs on 64 training sequences

The transformer is 6× faster at inference than the statistical baseline, which must compute windowed correlations.

6.3 Prior Experiment Results

Building on earlier experiments (Exp9, Exp10), we demonstrated:

Table 3: Alert ratios from prior experiments

Experiment	Clean Alerts	Attack Alerts	Ratio
Exp9 (Toy workload)	38	196	5.2×
Exp10 (LLM scale)	20	72	3.6×

These experiments used the statistical baseline with handcrafted thresholds. The transformer approach provides a more robust learned decision boundary.

7 Tetrahedral Geometry Enhancement

Following formal verification in Lean 4 [Moore, 2026], we explored enhanced sensor geometries. The original 3-oscillator (triangle) configuration yields 3 correlation pairs. A 4-oscillator (tetrahedron) configuration yields 6 pairs, theoretically providing $\sqrt{6/3} = \sqrt{2} \approx 1.41\times$ SNR improvement through additional degrees of freedom.

7.1 Experimental Validation

We extended the sensor to four oscillators with $r_D = 3.79$ (fourth vertex of the tetrahedron) and measured 6 correlations: $\rho_{AB}, \rho_{AC}, \rho_{AD}, \rho_{BC}, \rho_{BD}, \rho_{CD}$.

Table 4: Tetrahedral vs Triangle sensor comparison

Metric	Triangle (3 osc)	Tetrahedron (4 osc)	Improvement
Correlation pairs	3	6	2.0×
Baseline σ	0.061	0.054	1.13×
$\lvert\Delta\rho\rvert$ under attack	0.009	0.314	35×
Detection z-score	0.15σ	5.80σ	**38**×
3σ detectable	NO	**YES**	—

7.2 Signal Amplification Mechanism

The tetrahedral geometry exceeded theoretical predictions not through noise reduction, but through **signal amplification**. The 4th oscillator ($r = 3.79$) operates in a qualitatively different chaotic regime:

- Triangle $\Delta\rho$: -0.009 (barely detectable)
- Tetrahedron $\Delta\rho$: $+0.314$ (35× larger signal)

This occurs because the cross-correlations between oscillators at different r-values sample distinct regions of the PDN voltage response space. The additional oscillator acts as a *differential probe* sensitive to workload-induced power domain variations.

7.3 k-effective Stability

We validated the k-effective framework from coherence collapse analysis:

$$k_{\text{eff}} = \frac{k}{1 + \rho(k-1)} \tag{5}$$

where k is the number of correlation pairs.

Table 5: k-effective stability comparison

Configuration	k	Mean k_{eff}	Std	CV
Triangle	3	4.22	0.63	14.9%
Tetrahedron	6	8.04	2.98	37.1%

While the tetrahedron has higher coefficient of variation due to operating closer to the $\rho = -1/(k-1)$ singularity, its massively improved detection z-score (38×) makes it the preferred configuration.

8 Discussion

8.1 Why Transformers Outperform Statistical Methods

The statistical baseline computes instantaneous deviations from calibrated means. It cannot capture:

- **Temporal patterns:** Attack signatures may involve specific sequences of correlation changes
- **Multi-variate relationships:** Correlations between ρ_{AB}, ρ_{BC}, ρ_{AC} may carry information

- **Non-linear decision boundaries:** The separation between attack and clean may not be a simple threshold

The transformer's self-attention mechanism learns which temporal positions are most informative for classification, effectively discovering the "fingerprint" of each workload type.

8.2 Limitations

1. **Same-type attacks are invisible:** A matmul-heavy attack during matmul-heavy workload produces no detectable signal
2. **Heavy loads saturate the signal:** At very high GPU utilization, correlation approaches zero regardless of workload type
3. **Sample size:** 100 sequences is modest; larger datasets would improve generalization
4. **GPU-specific:** Fingerprints may differ across GPU architectures due to PDN differences

8.3 Connection to Coherence Collapse

This work instantiates the coherence collapse framework [Moore, 2026] in a new domain. The core insight—that correlation structure encodes system state—applies universally:

- **Batteries:** Cell correlation predicts degradation
- **Governance:** Institutional correlation predicts democratic backsliding
- **Microbiomes:** Species correlation predicts dysbiosis
- **GPUs:** Oscillator correlation predicts unauthorized workloads

9 Conclusion

We have demonstrated that transformer neural networks can detect unauthorized GPU workloads by learning temporal patterns in chaotic oscillator correlation time series. This approach achieves 75% accuracy with 0.76 F1 score, significantly outperforming statistical baselines that fail under realistic conditions.

The tetrahedral geometry enhancement provides a surprising 38× improvement in detection z-score—far exceeding the theoretical $\sqrt{2}$ prediction—by operating the 4th oscillator in a qualitatively different chaotic regime that acts as a differential probe for PDN perturbations. This enables single-sample detection at 5.8σ, crossing the 3σ threshold that the triangle configuration cannot achieve.

The method requires no external hardware, no performance counters, and minimal computational overhead. It exploits fundamental physical properties of GPU power delivery networks, making evasion difficult without reducing attack effectiveness.

Future work includes:

- Cross-GPU generalization studies across AMD, Intel, and older NVIDIA architectures
- Larger and more diverse attack datasets with adversarial evasion attempts
- Integration with production GPU monitoring systems and cloud provider infrastructure
- Multi-SM spatial arrays: deploying tetrahedral sensors across 4+ streaming multiprocessors for an expected $\sqrt{30/3} \approx 3.16\times$ additional SNR improvement
- Training transformers on the 6-dimensional tetrahedral correlation signal

Code and Data Availability

Source code is available at `https://github.com/cirisai/gpu_tamper_detection` under BSL 1.1 license (free for individuals, academics, nonprofits, and organizations under $1M revenue).

Acknowledgments

This work builds on the CIRIS ethical AI framework and coherence collapse analysis methodology developed at CIRIS L3C.

References

Moore, E. (2025). CIRISAgent open source ethical AI framework for accountable autonomy. Zenodo. `https://doi.org/10.5281/zenodo.17195221`

Moore, E. (2026). Coherence Collapse Analysis: A Universal Failure Mode in Complex Coordinating Systems. Zenodo. `https://doi.org/10.5281/zenodo.18142668`

Microsoft Security Blog. (2023). Cryptojacking: Understanding and defending against cloud compute resource abuse. `https://www.microsoft.com/en-us/security/blog/2023/07/25/cryptojacking-understanding-and-defending-against-cloud-compute-resource-abuse/`

Vectra AI. (2024). Cryptomining Attacks and How to Detect Them. `https://www.vectra.ai/modern-attack/attack-techniques/cryptomining`

Datadog. (2024). Detect cryptocurrency mining in your environment with Datadog Cloud SIEM. `https://www.datadoghq.com/blog/cryptomining-detection-rule/`

ARMO. (2024). Cloud Mining: Safeguard Your Cloud from Cryptominers. `https://www.armosec.io/blog/cloud-mining-security-threats/`

Giri, D. et al. (2024). GPU Side-Channel Attack Classification for Targeted Secure Shader Mitigation. *SN Computer Science*. `https://doi.org/10.1007/s42979-024-03514-9`

Maia, A. et al. (2022). Snooping the GPU via Magnetic Side Channel. *USENIX Security Symposium*.

Li, Y. et al. (2024). Exploiting HDMI and USB Ports for GPU Side-Channel Insights. *arXiv:2410.02539*.

Luo, C. et al. (2018). Power Analysis Attack of an AES GPU Implementation. *Journal of Hardware and Systems Security*.

Ngo, D. et al. (2020). Power side channel attack analysis and detection. *Proceedings of ICCAD*.

Drutarovsky, M. & Galajda, P. (2014). A Chaotic Ring oscillator based Random Number Generator. *IEEE Conference Publication*.

Koyuncu, I. et al. (2020). Chaotic Ring Oscillator Based True Random Number Generator Implementations in FPGA. *IEEE Conference Publication*.

Garipcan, A. & Ergun, S. (2018). Cryptanalysis of a Chaotic Ring Oscillator Based Random Number Generator. *IEEE Conference Publication*.

Biswas, H. et al. (2024). Enhanced logistic map with infinite chaos and its applicability in lightweight and high-speed pseudo-random bit generation. *Cybersecurity*.

Buscarino, A. et al. (2019). Enhancing the sensitivity of a chaos sensor for Internet of things. *Internet of Things*.

Tuli, S. et al. (2022). TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data. *VLDB*.

Shang, L. et al. (2024). An Efficient Anomaly Detection Method for Industrial Control Systems: Deep Convolutional Autoencoding Transformer Network. *International Journal of Intelligent Systems*.

Srivastava, A. et al. (2023). BTAD: A binary transformer deep neural network model for anomaly detection in multivariate time series data. *Advanced Engineering Informatics*.

Ahmed, M. et al. (2025). Evaluating large transformer models for anomaly detection of resource-constrained IoT devices for intrusion detection system. *Scientific Reports*.

Google Cloud. (2024). Security Command Center Cryptomining Protection. `https://cloud.google.com/security-command-center/cryptomining-protection-program`

Leng, J. et al. (2014). GPUVolt: Modeling and Characterizing Voltage Noise in GPU Architectures. *ISLPED*.