First, sequences of system calls from different files were gathered in this model and classified as normal behavior (0) or attack behavior (1). The sequences were subsequently transformed into feature vectors via TF-IDF (Term Frequency–Inverse Document Frequency), which captured 1-gram, 2-gram, and 3-gram patterns.

With this approach, the model could recognize individual system calls and also comprehend short sequences of calls, which allowed it to capture more significant behavioral patterns.

The sparse feature vectors produced were input into an XGBoost classifier. This classifier is ideal for high-dimensional sparse data and can efficiently model complex relationships using boosted decision trees.

The model managed to reach an accuracy of about 97.2% by learning subtle differences in syscall usage and their short-term orderings.

By combining semantic pattern extraction through n-gram TF-IDF with regularized learning of XGBoost, the model achieved both accuracy and generalizability for host-based intrusion detection using system call traces.

This approach, in contrast to published papers, focuses on simple but effective text-based feature engineering and lightweight supervised learning, while many other methods incorporate deep semantic modeling or hybrid detection techniques.

For instance, the paper titled “Dynamic System Call Anomaly Detection” (2022) merges signature-based detection with behavior modeling through semantic representations of system calls. This process frequently necessitates more intensive customized feature extraction or deep learning techniques.

Likewise, the work titled “Enhanced HIDS Using Syscall Traces” (2023) employs manually created lightweight features along with basic classifiers, which leads to a lower accuracy of approximately 90.48%.

Our model, in contrast, views syscall sequences as textual data and extracts significant short patterns using TF-IDF n-grams. It utilizes a robust ensemble model (XGBoost) without the need for manual feature design or hybrid systems.

This ensures that your pipeline remains straightforward, quick to implement, and scalable while achieving a competitive performance (~97%) without the need for complex behavior modeling. This practicality is especially beneficial for real-world deployments where efficiency and interpretability are crucial.

| **Aspect** | **Our Model (TF-IDF + XGBoost)** | **Dynamic Syscall Anomaly Detection (2022)** | **Enhanced HIDS Using Syscall Traces (2023)** |
| --- | --- | --- | --- |
| **Feature Extraction** | TF-IDF n-gram (1–3 grams) | Semantic syscall representation + signature | Lightweight handcrafted features |
| **Learning Method** | XGBoost classifier | Hybrid (signature + behavior modeling) | Simple classifiers (e.g., SVM, Decision Tree) |
| **Complexity** | Moderate (simple pipeline, scalable) | High (needs semantic modeling) | Low (basic feature design) |
| **Accuracy** | ~97.0% | 99.0% | 90.48% |
| **Strength** | Fast, interpretable, scalable | Highly accurate but complex setup | Very lightweight, easy to implement |
| **Weakness** | May miss very deep behavior patterns | Heavy computational and design overhead | Lower detection accuracy |

A screenshot of a computer screen

AI-generated content may be incorrect.

A graph with purple squares

AI-generated content may be incorrect.