令和 4 年度 特別研究報告書要旨

トレースポイントを用いた Linux カーネルの動作の監視

T190524　林　彩菜
指導教員： 三好 力 教授，芝 公仁 助教

1　はじめに

一般的に，オペレーティングシステムは，潜在的なバグや脆弱性が存在し，機能の追加や修正によって新たなバグや脆弱性が発生する可能性がある．オペレーティングシステムの状態把握や異常検知を可能にするためには，オペレーティングシステムの主要なコンポーネントである Linux カーネルの挙動を監視することが最も重要である [1]．そこで本論文では，カーネルのデバッグ用の機能であるトレースポイントを用いて，カーネル内での実際の動作を監視し，現在の状態を把握するシステムを作成すること目的する．

2　システムの構成と処理

図 1 にシステムの構成を示す．この構成は，カールトレーサからトレースポイントフックモジュールを呼び出し，一定数のトレースポイントをユーザ空間にログとして保存する．カーネルトレーサは python スクリプトで，トレースポイントの表の作成やカーネルモジュールの呼び出し，データを保存するログファイルの作成を行う．トレースポイントフックモジュールは Linux カーネルモジュールで，イベント発生時毎にそのトレースポイントのデータとしてトレースポイントの識別子とタイムスタンプを格納する．

まず，取得した一部を除くトレースポイントに識別子を割り当て，トレースポイント名とともにトレースポイント表として作成する．次に，このトレースポイント表を用いて，イベント発生時のトレースポイントのデータを取得する．そして，終了条件を満たした後，取得したトレースポイントのデータであるタイムスタンプおよびトレースポイント識別子をログファイルに書き込み，システムを終了する．

3　評価

本論文では，オーバヘッドによるシステムの評価を行う．スクリプトを実行する際に必要なオープンファイル数の上限値を最大の 104856 に設定し，本システムを使用している時と使用していない時の Linux カーネルパッケージのビルドの時間

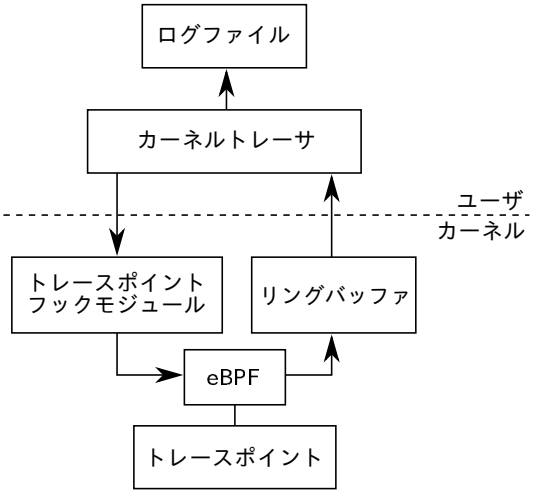

図 1　システムの構成

表 1　測定時間の比較

項目	測定時間 (s)
システム使用時	2303.36
システム未使用時	2054.47

を測定し，評価を行った．表 1 に，各測定時間を示す．

システム使用時とシステム未使用時の時間の差は，248.89 秒となった．これは，およそ 1.12%の増加であり，オーバヘッドはあまり大きくないと考えられる．

4　おわりに

本論文では，トレースポイントを用いて Linux カーネルの動作を監視するシステムについて述べた．本システムは，いくつかの環境で Linux カーネルがどのような動きをしているのか，トレースポイントを用いることで，具体的にどのような処理が多いのかを明らかにすることができた．これによって，Linux カーネルの動作の監視し，状態を把握することが実現される．

参考文献

[1] Gebai, M. and Dagenais, M. R.: Survey and Analysis of Kernel and Userspace Tracers on Linux: Design, Implementation, and Overhead, *ACM Computing Surveys*, Vol. 51, pp. 1–33 (2019).