Detecting Intrusions on Windows Operating Systems by Monitoring System Services

学部4年生
島本 大輔
2005年2月8日

現在のセキュリティ事情

- □ インターネットの成長による Attack による危険性の増大
  - ■ ウィルス、不正侵入、スパイウェア、など

⇒ Intrusion Detection System (IDS) の重要性

Intrusion Detection System(IDS)

- □ 侵入検知システム
  - ■ Signature based ＝ パターンマッチ
    - □ アンチウィルスソフト、など
  - ■ Anomaly based ＝ 異常な動作を検知

既存の IDS

- □ 既存のWindows 用 IDS
  - ■ Signature based が主流
    ＝新種への対応が困難
- □ Linux、UNIX における IDS の研究
  - ■ 新種への対応を含めた先進的な研究が多数

Linux、UNIX における IDS

- □ プロセスの様々な情報を監視
  - ■ Forrest らの System Callの監視
  - ■ Feng らのスタックの監視
  - ■ Sekar らのプログラムカウンタ (PC) の監視
- □ これらのWindows への応用が少ない
  - ■ ブラックボックスなOS
  - ■ 公式に提供されているモジュールでは不十分

モチベーション

- □ 「Linux、UNIX における先進的な IDS の研究を Windows へ応用したい」

⇒ Windows における System Call などの情報を使って異常を検出

研究内容

- ☐ IDS への第一歩
  - Windows 版 System Call である System Service を Intercept
  - System Service のパターンを調査

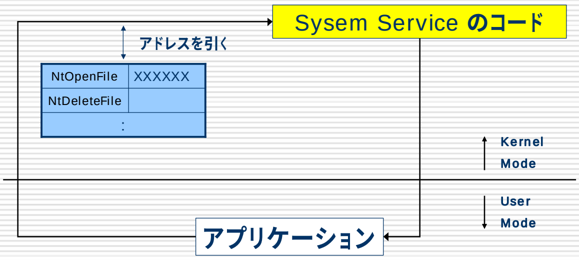
System Service

- ☐ Windows の根本的な機能を提供
  - ファイル操作、レジストリ操作、プロセス操作、など

例：NtWriteFile･･･ファイルへの書き込みはすべて
これを利用

- ☐ UNIX 系 OS の System Call に対応

System Service の動作



Sysem Service のコード

アドレスを引く

NtOpenFile	XXXXXX
NtDeleteFile	
:	

Kernel Mode

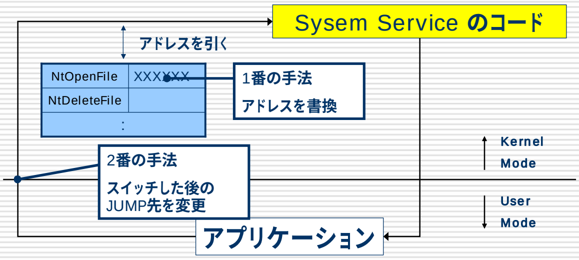
User Mode

アプリケーション

System Service の Interception

- ☐ 2通り存在
  1. System Serviceのアドレステーブル（System Service Descriptor Table）を書換
     - ☐ System Call Table の書き換えと同等
  2. Kernel mode へスイッチする瞬間に Intercept
     - ☐ Windows 2000 以前では int 2e
     - ☐ Windows XP 以降は SYSENTER 命令

System Service の Interception



Sysem Service のコード

アドレスを引く

NtOpenFile	XXXXXX
NtDeleteFile	
:	

1番の手法
アドレスを書換

2番の手法
スイッチした後の
JUMP先を変更

Kernel Mode

User Mode

アプリケーション

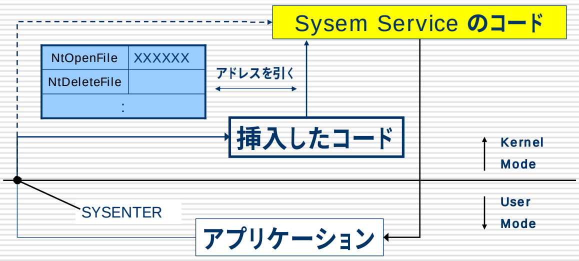
System Service の Interception

- ☐ 2番の SYSENTER 命令を利用した Interception を採用
  - 既存研究が少数
  - 監視する System Service の追加･削除が容易
  - 一箇所において監視可能

※SYSENTER 命令

- ☐ Fast System Call･･･System Call への遷移を
1命令で完了

System Service の Interception



実装内容

- □ Interceptionはデバイスドライバとして動作
- □ 記録の出力のために別途、User Mode プログラムを開発
  - ■ System Service の記録を出力

System Service のパターンの取得

- □ 取得方法
  - ■ VMWare上でウィルスと今回のプログラムを実行
- □ 様々なウィルスのパターンを取得
  - ■ WORM_MUMU.B, WORM_NETSKY.Z, WORM_BEGLE.AT, WORM_SASSER.B など
  - ■ 今後、より多くのパターンの調査が必要

出力の例 (WORM_MUMU.B)

```
0112 0112 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 0074
0074 0074 0074 0074 0074 0074 0074 0074 0074 0074 00b7
```

0112 = NtWriteFile, 0074 = NtOpenFile, 00b7 = NtReadFile

まとめ

- □ モチベーション
  - ■ Windows における先進的なIDSの実装
- □ 今回の研究内容
  - ■ Interception の手法を調査・実装
  - ■ ウィルスなどのパターンを取得
- □ 今後の予定
  - ■ IDS への拡張