The Center for Cyber Defenders

Expanding computer security knowledge

SAND2016-7010C

Characterization of System Calls

Micah Bushouse, North Carolina State University

Project Mentors: Joe Ingram, Org. 9525; Mike Smith, Org. 5621

Problem Statement:

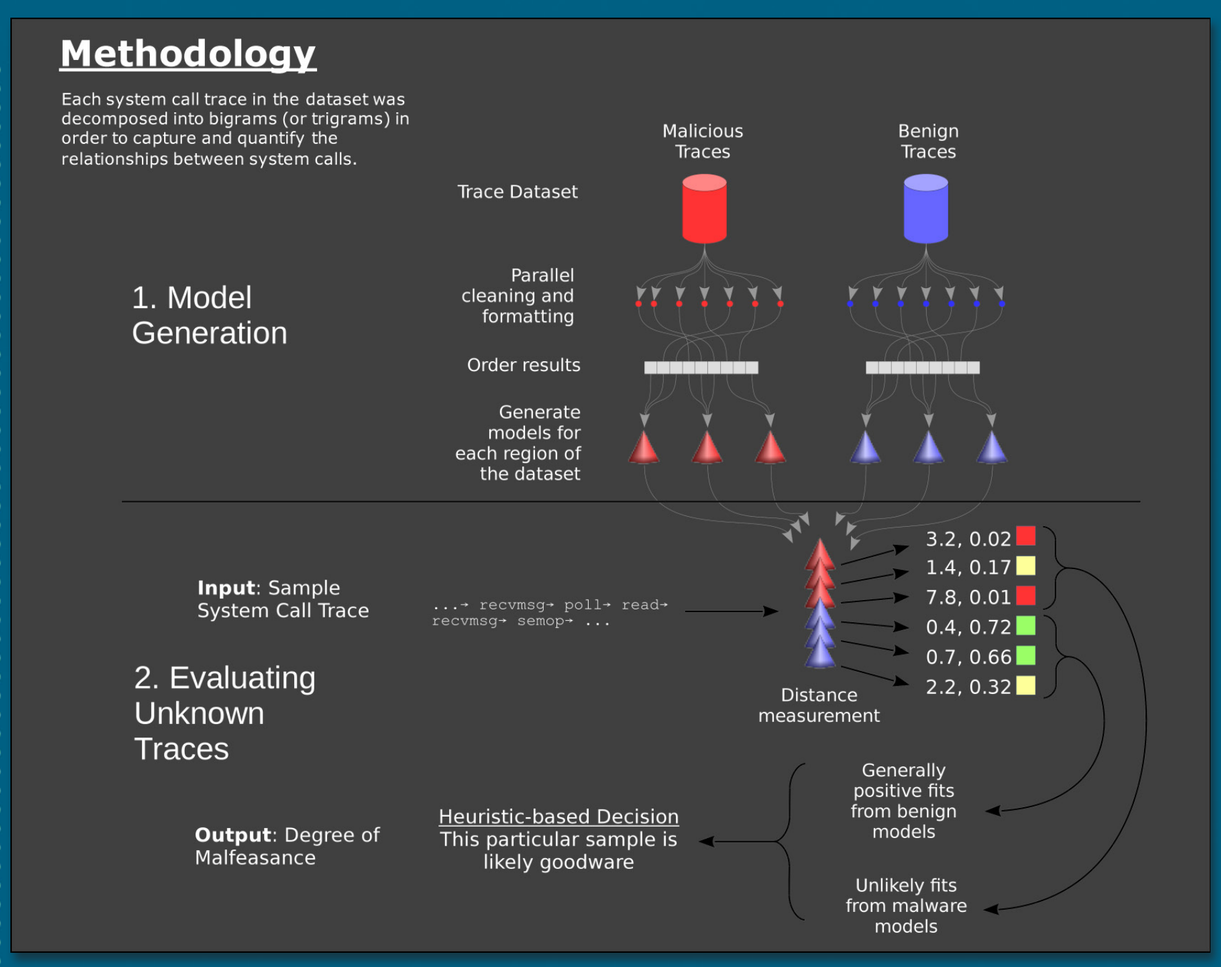
- Can we detect malicious activity using statistical models generated from a system call (syscall) trace dataset?

Objectives:

- Explore the use of a distance metric test to classify malicious activity. In this study, the Kolmogorov–Smirnov (KS) test was used as the distance metric.
- Develop techniques to clean and visualize this dataset for further work.

Methodology

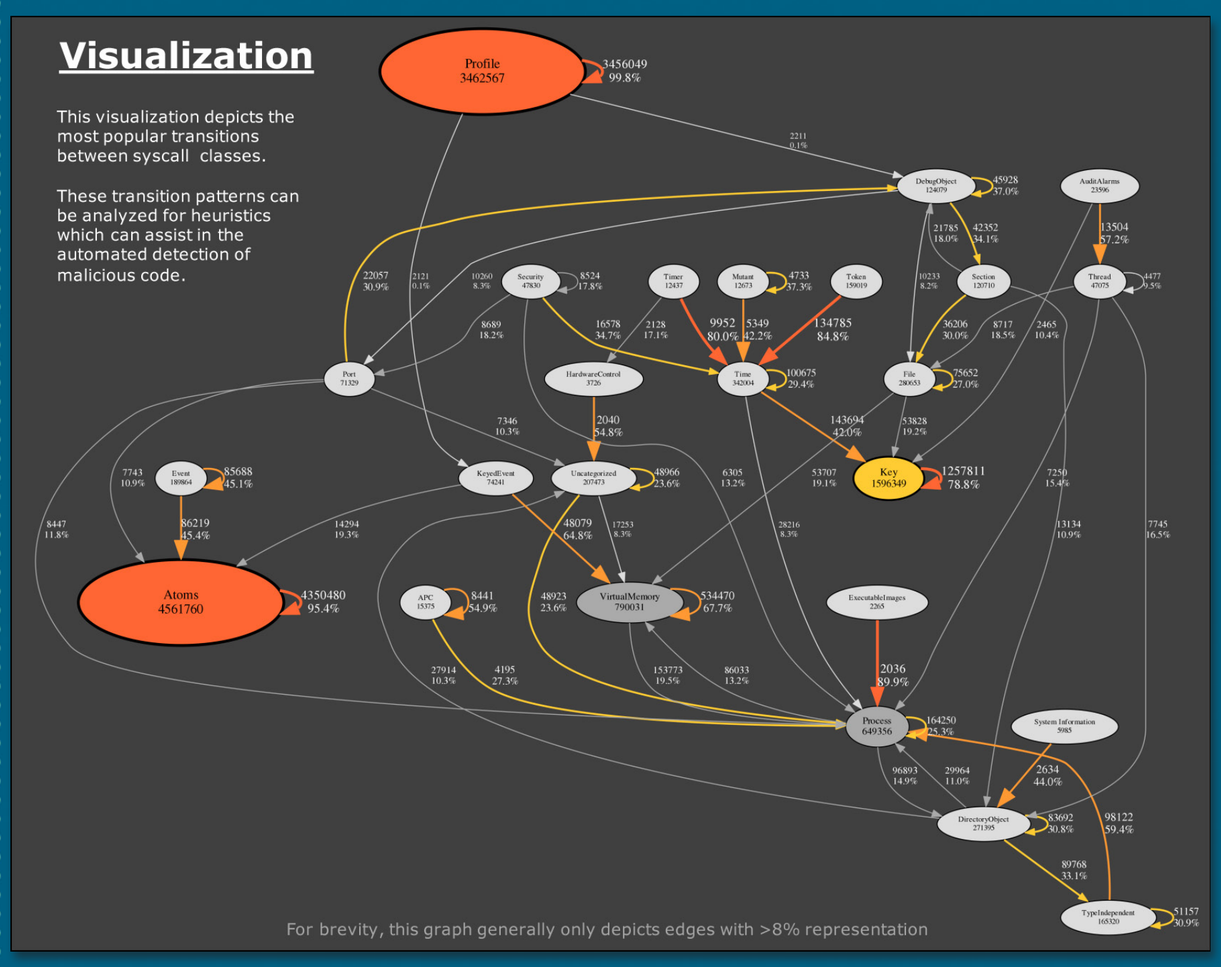
Each system call trace in the dataset was decomposed into bigrams (or trigrams) in order to capture and quantify the relationships between system calls.



Visualization

This visualization depicts the most popular transitions between syscall classes.

These transition patterns can be analyzed for heuristics which can assist in the automated detection of malicious code.



Results:

n = 574		Predicted Positive	Predicted Negative
Actual	Positive	**205** TP	**42** FN
	Negative	**102** FP	**227** TN

Precision 0.668
Recall 0.830

Parameters:

- Syscall relationships were captured in bigrams.
- System calls were binned into 30 categories organized by operating system primitive.
- 40 unsorted models were created for each label (benign and malicious).
- A KS test was performed between the sample and each model, and the closest three models from each label were compared to classify the sample.

Conclusion:

- Initial results were fair but demonstrate the potential of this approach.
- Taken as is, this method could be used in an ensemble of heuristics.

What are Syscalls?

System calls are the **standardized, well-defined programmatic pathways** for programs to interact with the operating system.

Programs use system calls **to request and manipulate computer resources** controlled by the operating system, including files and connections.

The **pattern** in which a program invokes system calls can be captured, and a program's behavior can be inferred by the system calls it uses.

Windows 7 uses over 700 system calls.

System Call Categories

- **Process Control**
  Load, execute, end, abort, create process, terminate process get/set process attributes
- **File management**
  create file, delete file, open, close, read, write, reposition, get/set file attributes
- **Device Management**
  request device, release device, read, write, reposition, get/set device attributes
- **Information Maintenance**
  get/set time or date, get/set system data, get/set process, file, or device attributes
- **Communication**
  create, delete communication connection, send, receive messages, transfer status

Capturing Syscall Transitions

Each system call trace in the dataset was decomposed into bigrams (or trigrams) in order to capture and quantify the relationships between system calls.

Syscall Trace	Resulting Bigrams	Resulting Trigrams
poll	(poll, recvmsg)	(poll, recvmsg, recvmsg)
recvmsg	(recvmsg, recvmsg)	(recvmsg, recvmsg, poll)
recvmsg	(recvmsg, poll)	(recvmsg, poll, read)
poll	(poll, read)	(poll, read, recvmsg)
read	(read, recvmsg)	(read, recvmsg, read)
recvmsg	(recvmsg, read)	(recvmsg, read, semop)
read	(read, semop)	(read, semop, nanosleep)
semop	(semop, nanosleep)	(semop, nanosleep, waitpid)
nanosleep	(nanosleep, waitpid)	(nanosleep, waitpid, _newselect)
waitpid	(waitpid, _newselect)	(waitpid, _newselect, gettid)
_newselect	(_newselect, gettid)	(_newselect, gettid, semop)
gettid	(gettid, semop)	(gettid, semop, ...)
semop	(semop, ...)	(semop, ..., ...)
...		

t ↓

Dataset and Performance

	Benign	Malicious
Traces	27k	11k
Bigrams	98.9M	599.4M
Unique Bigrams	3701	4604
Model Build Time	511 sec	782 sec
Average Model Size	2231	
Dataset/Model Ratio*	Approx. 13,000:1	
Time to Parse and Classify ~570 Samples	1267 sec (2.2/sec/sample)	

*: Models were constructed as a scaled representation of their segment of the dataset. The scaling algorithm used the quotient between the most frequent bigram and a fixed value (1000) to scale each remaining bigram. After being scaled in this way, less represented bigrams with a scaled value of <1.0 were clipped.

python™ matplotlib NumPy SciPy GRAPHVIZ

System Call Categories are attributed to: https://en.wikipedia.org/wiki/System_call

Sandia National Laboratories LOCKHEED MARTIN

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-84AL8500. SAND2016-XXXXC.

U.S. DEPARTMENT OF ENERGY NNSA National Nuclear Security Administration