On the Forensic Validity of Approximated Audit Logs

Noor Michael, Jaron Mink, Jason Liu, Sneha Gaur, Wajih Ul Hassan, and Adam Bates

University of Illinois at Urbana-Champaign
Audit Logs are Invaluable
Audit Logs are Invaluable

- Records history of executed events
  - Kernel-level frameworks track application syscalls
Audit Logs are Invaluable

- Records history of executed events
  - Kernel-level frameworks track application syscalls

- 75% of analysts [1] believe logs are the most important resource when investigating threats

Audit Logs are Invaluable … but Burdensome

- Records history of executed events
  - Kernel-level frameworks track application syscalls
- 75% of analysts [1] believe logs are the most important resource when investigating threats


Audit Log Reduction Techniques

Insight: The entire audit log is not often required
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
● not needed for investigation goal
● redundant
● reasonably approximated
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine where process A sent data
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine where process A sent data

Original Log

```plaintext
1: <Proc A, t_001, send, server.com>
...
99: <Proc A, t_099, send, server.com>
```
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine where process A sent data

Original Log

1: <Proc A, t_001, send, server.com>
...
99: <Proc A, t_099, send, server.com>

Approximated Log

1: <Proc A, t_0XX send, server.com>
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine where process A sent data

Original Log

```
1: <Proc A, t_001, send, server.com>
...
99: <Proc A, t_099, send, server.com>
```

Approximated Log

```
1: <Proc A, t_0XX send, server.com>
```

The same conclusion is reached with either log
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Original Log:

1: <Proc A, t_001, send, server.com>
... 
99: <Proc A, t_099, send, server.com>

Approximated Log:

1: <Proc A, t_0XX send, server.com>
Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine whether Proc A was using a covert timing channel

Original Log

1: <Proc A, t_001, send, server.com>
...
99: <Proc A, t_099, send, server.com>

Approximated Log

1: <Proc A, t_0XX send, server.com>

Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine whether Proc A was using a covert timing channel

Original Log

1: <Proc A, t_001, send, server.com>
... 
99: <Proc A, t_099, send, server.com>

Approximated Log

1: <Proc A, t_0XX send, server.com>

Conclusions may differ!

Audit Log Reduction Techniques

Insight: The entire audit log is not often required

Information may be:
- not needed for investigation goal
- redundant
- reasonably approximated

Investigation Goal: Determine whether Proc A was using a covert timing channel

Original Log

1: <Proc A, t_001, send, server.com>
...
99: <Proc A, t_099, send, server.com>

Approximated Log

1: <Proc A, t_0XX send, server.com>

Conclusions may differ!

How much information is kept for arbitrary goals under different threat models?

Formalizing Forensic Metrics
Formalizing Forensic Metrics

**Provenance Graph**

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

**Provenance Graph**

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

**Provenance Graph**

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

**Provenance Graph**

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

Provenance Graph

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

**Provenance Graph**

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics

Provenance Graph

Nodes: System Objects

Edges: Causal Events
Formalizing Forensic Metrics
Formalizing Forensic Metrics

**Lossless**

**Threat Model:**
Diverges from system level abstractions

**Preserves:**
All Information

Formalizing Forensic Metrics

Lossless

Threat Model:
Diverges from system level abstractions

Preserves:
All Information

Formalizing Forensic Metrics

**Lossless**

**Threat Model:**
*Diverges* from system level abstractions

**Preserves:**
All Information

## Formalizing Forensic Metrics

<table>
<thead>
<tr>
<th>Lossless</th>
<th>Causality-Preserving (based on Xu et. al.¹)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Threat Model:</strong></td>
<td><strong>Threat Model:</strong></td>
</tr>
<tr>
<td>Diverges from system level abstractions</td>
<td>Abides by system level abstractions</td>
</tr>
<tr>
<td><strong>Preserves:</strong></td>
<td><strong>Preserves:</strong></td>
</tr>
<tr>
<td>All Information</td>
<td>Information flow</td>
</tr>
</tbody>
</table>

Formalizing Forensic Metrics

**Lossless**

**Threat Model:**
Diverges from system level abstractions

**Preserves:**
All Information

**Causality-Preserving**

*(based on Xu et. al.)*

**Threat Model:**
Abides by system level abstractions

**Preserves:**
Information flow

---

Formalizing Forensic Metrics

<table>
<thead>
<tr>
<th>Lossless</th>
<th>Causality-Preserving (based on Xu et. al.)</th>
<th>Attack-Preserving</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threat Model: Diverges from system level abstractions</td>
<td>Threat Model: Abides by system level abstractions</td>
<td>Threat Model: Abides by system level abstractions</td>
</tr>
<tr>
<td>Preserves: All Information</td>
<td>Preserves: Information flow</td>
<td>Preserves: Uniquely Malicious information flow</td>
</tr>
</tbody>
</table>

Formalizing Forensic Metrics

**Lossless**

**Threat Model:** Diverges from system level abstractions  
**Preserves:** All Information

**Causality-Preserving**  
(based on Xu et. al.¹)

**Threat Model:** Abides by system level abstractions  
**Preserves:** Information flow

**Attack-Preserving**

**Threat Model:** Abides by system level abstractions  
**Preserves:** Uniquely Malicious information flow

---

Formalizing Forensic Metrics

Lossless
- benign.com
- mal.com
- recv
- recv
- write x3
- fork
- send
- read x2
- benign.txt
- mal.exe
- exfiltration.com
- passwords.txt

Causality-Preserving
- benign.com
- mal.com
- recv
- recv
- write
- fork
- send
- read
- benign.txt
- mal.exe
- exfiltration.com
- passwords.txt

Attack-Preserving
- benign.com
- mal.com
- recv
- recv
- fork
- send
- read
- benign.txt
- mal.exe
- exfiltration.com
- passwords.txt
Formalizing Forensic Metrics

Lossless

Causality-Preserving

Attack-Preserving

\[ \text{Reduction}_i(G) = G_i \]
Formalizing Forensic Metrics

\[ \text{Ideal}_m(G) = G_m \]

\[ \text{Reduction}_i(G) = G_i \]
Formalizing Forensic Metrics

\[ \text{Reduction}_i(G) = G_i \]
\[ \text{Ideal}_m(G) = G_m \]
\[ E(G) = \{ e \} \]
Formalizing Forensic Metrics

**Lossless**

- `benign.com` → `recv`
- `mal.com` → `recv`
- `write x3`
- `firefox.exe`
- `mal.exe`
- `send` → `read x2`
- `benign.txt`
- `passwords.txt`
- `exfiltration.com`

**Causality-Preserving**

- `benign.com` → `recv`
- `mal.com` → `recv`
- `write`
- `firefox.exe`
- `mal.exe`
- `send` → `read`
- `benign.txt`
- `passwords.txt`
- `exfiltration.com`

**Attack-Preserving**

- `benign.com` → `recv`
- `mal.com` → `recv`
- `fork`
- `firefox.exe`
- `mal.exe`
- `send` → `read`
- `benign.txt`
- `passwords.txt`
- `exfiltration.com`

\[
\text{Reduction}_i(G) = G_i
\]

\[
\text{Ideal}_m(G) = G_m
\]

\[
E(G) = \{ e \}
\]

\[
\text{Forensic Validity}_{i,m} = \frac{|E(G_i) \cap E(G_m)|}{|E(G_m)|}
\]
LogApprox
LogApprox
LogApprox
LogApprox

Diagram:
- benign.com
- mal.com
- firefox.exe
- mal.exe
- exfiltration.com
- Cache/11/page1.html
- Cache/12/page2.html
- passwords.txt

Connections:
- benign.com to firefox.exe: recv
- mal.com to firefox.exe: recv
- firefox.exe to mal.exe: r/w
- mal.exe to exfiltration.com: send
- mal.exe to passwords.txt: read
LogApprox

Diagram showing interactions between processes and files, including
- benign.com
- mal.com
- /lib/libc.so.7
- firefox.exe
- Cache/13/page3.html
- mal.exe
- exfiltration.com
- passwords.txt

Actions indicated in the diagram:
- recv
- recv
- read x 3
- write x 3
- fork
- read x 3
- send
- read
LogApprox
LogApprox

Reduction Opportunities

- Most system events are file IO events!
  - Related files unable to be causally reduced
LogApprox

Reduction Opportunities
● Most system events are file IO events!
  ○ Related files unable to be causally reduced

LogApprox Reduction:
● Coalesce repetitive IO activity via regexes
Filepaths for Firefox.exe
/Cache/11/page1.html
/Cache/12/page2.html
/Cache/13/page3.html
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2

LogApprox
LogApprox

**Group 1:**
/Cache/11/page1.html

**Filepaths for Firefox.exe**

/Cache/12/page2.html
/Cache/13/page3.html
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2
LogApprox

**Group 1:**
/Cache/11/page1.html

**Filename Similarity:** A
Levenshtein Edit Distance

**Path Distance:** β
Number of different directories

**Filepaths for Firefox.exe**

/Cache/12/page2.html
/Cache/13/page3.html
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2
Group 1:
/Cache/11/page1.html
/Cache/12/page2.html
/Cache/13/page3.html

Group by:
Filename Similarity: A
Levenshtein Edit Distance

Path Distance: β
Number of different directories

Filepaths for Firefox.exe
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2
LogApprox

**Group 1:**
/Cache/11/page1.html
/Cache/12/page2.html
/Cache/13/page3.html

**Group 2:**
/lib/libc.so.1

**Filepaths for Firefox.exe**

/lib/libc.so.6
/lib/libc.so.7
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2

**Group by:**

- **Filename Similarity:** A
- Levenshtein Edit Distance
- **Path Distance:** β
- Number of different directories
LogApprox

**Group 1:**
/Cache/11/page1.html
/Cache/12/page2.html
/Cache/13/page3.html

**Group 2:**
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7

**Group 3:**
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2

**Group by:**

**Filename Similarity:** A
Levenshtein Edit Distance

**Path Distance:** β
Number of different directories
Group 1:
/CACHE/11/page1.html
/CACHE/12/page2.html
/CACHE/13/page3.html
-----------------------------------
/CACHE/*/page*

Group 2:
/lib/libc.so.1
/lib/libc.so.6
/lib/libc.so.7
-----------------------------------
/lib/libc.so.*

Group 3:
/lib64/libQt3t.so.1
/lib64/libQt3t.so.1.1
/lib64/libQt3t.so.1.2
-----------------------------------
/lib64/libQt3t.so.1*

Group by:
Filename Similarity: A
Levenshtein Edit Distance
Path Distance: β
Number of different directories
Firefox IO Templates:

/Cache/*/*page*

/lib/libc.so.*

/lib64/libQt3t.so.1*
Firefox IO Templates:

/Cache/*/page*
/lib/libc.so.*
/lib64/libQt3t.so.1*
Firefox IO Templates:

/Cache/*/page*
/lib/libc.so.*
/lib64/libQt3t.so.1*

LogApprox

benign.com
mal.com

recv
recv
recv

write
read

read
fork

read
read

send

exfiltration.com

passwords.txt

firefox.exe

/lib/libc.so.*
/lib64/libQt3t.so.1*

mal.exe

Cache/*/page*
LogApprox

Properties

- benign.com
- mal.com
- /lib/libc.so.*
- /lib64/libQt3t.so.1*
- firefox.exe
- Cache/*page*
- mal.exe
- exfiltration.com
- passwords.txt
LogApprox

Properties

- Only reduces repetitive \textit{local} file IO
LogApprox

Properties

- Only reduces repetitive *local* file IO
- IO is only ever *approximated*
LogApprox

Properties

- Only reduces repetitive local file IO
- IO is only ever approximated

LogApprox can receive high reduction rates while preserving anomalous behavior!
Evaluation against Exemplar Reduction Techniques
Evaluation against Exemplar Reduction Techniques

Causality-Preserving Reduction by Xu et. al.
Evaluation against Exemplar Reduction Techniques

Causality-Preserving Reduction
by Xu et. al.

LogGC
by Lee et. al.
Evaluation against Exemplar Reduction Techniques

Causality-Preserving Reduction
by Xu et. al.

LogGC
by Lee et. al.

Full and Source Dependence Preserving Reduction
by Hossain et. al.
Evaluation against Exemplar Reduction Techniques

Causality-Preserving Reduction
by Xu et. al.

LogGC
by Lee et. al.

Full and Source Dependence Preserving Reduction
by Hossein et. al.

Details of each algorithm in the paper!
(and within their respectively published papers!)
Curated set of real-world vulnerabilities and exploits:

- unrealircd\(^1\) : IRC Server
- vsftpd\(^2\) : FTP Server
- webmin\(^3\) : System Configuration Tool
- Wordpress\(^4\) : Content Management System
- PHP Webshell\(^5\) : Generic Web Server
- Firefox\(^6\) : Web Browser

Results

![Graph showing the relationship between Log Reduction (%) and Forensic Validity (%). The graph includes a line labeled 'Worst-case.'](image)
Results

Lossless Forensics
Results

Causality-Preserving Forensics
(all information flow)
Results

Causality-Preserving Forensics
(all information flow)
Results

Causality-Preserving Forensics
(all information flow)
Results

Causality-Preserving Forensics
(all information flow)
Results

Causality-Preserving Forensics
(all information flow)
Results

Causality-Preserving Forensics
(all information flow)

Attack-Preserving Forensics
(uniquely malicious information flow)
Results

Causality-Preserving Forensics
(all information flow)

Attack-Preserving Forensics
(uniquely malicious information flow)
Results

Causality-Preserving Forensics (all information flow)

Attack-Preserving Forensics (uniquely malicious information flow)
Takeaways
Validity of reduced logs should **not** be based on anecdotal studies

- Depends on **task** and **threat model**
- Providing a continuous metric for arbitrary queries is a step in the right direction
Takeaways

Validity of reduced logs should **not** be based on anecdotal studies
- Depends on **task** and **threat model**
- Providing a continuous metric for arbitrary queries is a step in the right direction

Reduction techniques can be tailored to specific tasks and threats
- Tasks: Source and Full Dependency Preserving
- Threat Models: **LogApprox**
Thank You!