Determining the Integrity of Remote System Call Streams

Jonathon Giffin
University of Wisconsin
Overview

- Runtime Monitoring
- Model Construction
- Binary Rewriting
- Model Precision
Countering Remote Attacks

- **Goal:** Even if an intruder can see, examine, and fully control the remote job, no harm can come to the local machine.
- **Key technology:** Static analysis of binary code.
Countering Remote Attacks

User Job

Analyzer

Checking Shadow

Modified User Job
Runtime Monitoring

Checking Shadow

Modified User Job

Call 2
Model Construction

- User Job
- Analyzer
  - Checking Shadow
  - Modified User Job
- Binary Program
- Control Flow Graphs
- Local Automata
- Global Automata
Control Flow Graph Generation

function( int a ) {
    if( a < 0 ){
        read( 0, 15 );
        foo();
    } else {
        read( a, 15 );
        close( a );
    }
}
Control Flow Graph Translation

CFG ENTRY

if

call read

call foo

return

return

CFG EXIT

read

read

foo

close
Non-deterministic Finite Automata (NFA)

- Structure
  - States
  - Labeled edges between states
- Edge labels are input symbols - call names
- Path to any accepting state defines valid sequence of calls
Control Flow Graph Translation

CFG ENTRY

if

call read

call foo

call close

return

CFG EXIT

read

read

foo

close
Interprocedural Model Generation

B

A

5 August 2003

WiSA - Jonathon Giffin
Possible Paths

B

foo

write

close

ε

A

read

read

close

ε
Possible Paths

B

foo

write

ε

close

ε

A

read

close

read
Impossible Paths

foo

write

read

close

B

A
Impossible Paths

B

foo

write

close

ε

ε

ε

read

read

close

5 August 2003

WiSA - Jonathon Giffin
Push-down Automata (PDA)

- NFA has interprocedural imprecision
  - Does not model state of call stack
- Language of application is context-free
- Solution:
  - Splice automata using PDA edges
  - Introduces new challenges
Left Recursion Challenge

```javascript
function last( Node n ) {
  if( n.next == null )
    write( fd, n.name );
  else
    last( n.next );
}
```
Left Recursion Challenge

- Possible solutions
  - Top down parsing
  - Esparza, Hansel, Rossmanith, Schwoon Algorithm
  - Bounded Stack PDA
  - PDA / NFA Hybrid
Hard Control Flow Issues

- Indirect calls
  - Solution: slice entire program
- Long jmps
  - Solution: assume all set jmps possible
- Indirect jumps
  - Solution: recover jump tables; slice program
- Signals
  - Solution: out-of-band data
Rewriting User Job

User Job

Analyzer

Checking Shadow

Modified User Job

Binary Program

Rewritten Binary
Call Site Renaming

- Give each monitored call site a unique name
- Captures arguments
- Obfuscation
- Limits attack call set
- Reduces nondeterminism

```cpp
function( int a ) {
    if( a < 0 ) {
        read( 0, 15 );
        foo();
    } else {
        read( a, 15 );
        close( a );
        _638();
        _83( a );
        _1920( a );
    }
}
```
Call Site Renaming

- Give each monitored call site a unique name
- Captures arguments
- Obfuscation
- Limits attack call set
- Reduces nondeterminism
Prototype Implementation

• Simulates remote execution environment
• Models supported:
  – NFA, PDA, Bounded PDA, Hybrid
• Optimizations:
  – Epsilon reduction, minimization, dependency calculation, automata inlining, dead automata removal
Analysis Metric

• Experiments evaluate model precision
• Average branching factor metric
Linux glibc socket model

entry

socket

exit
Solaris 8 libc socket model
Analysis Levels

- Two levels of Solaris 8 libc analysis
  - 1: Calling structure flattened & simplified
  - 2: True code
- Three levels of program analysis
  - 1: No renaming; no argument capture
  - 2: Renaming only
  - 3: Renaming & argument capture
Analysis - GNU Finger

1. Flatten & Simplify
2. True Code

Average Branching Factor

1. No Renaming, No Args
2. Renaming Only
3. Rename & Arguments

libc Analysis

1. Flatten & Simplify
2. True Code

0.007
Analysis - Procmail

1. Flatten & Simplify
2. True Code

libc Analysis

Average Branching Factor

1. No Renaming, No Args
2. Renaming Only
3. Rename & Arguments
Contributions

• Binary modeling
• Model optimizations
• Obfuscation
Determining the Integrity of Remote System Call Streams

Jonathon Giffin
University of Wisconsin