Shellcode Detection in IPv6 Networks with HoneydV6

Sven Schindler

Universität Potsdam

Potsdam University
Institute for Computer Science
Operating Systems and Distributed Systems

Vienna, August 30, 2014
Outline

1. Introduction
2. Shellcode detection and analysis
3. Honeypot shellcode detection extension
4. Evaluation
5. Summary
Outline

1. Introduction
2. Shellcode detection and analysis
3. Honeypot shellcode detection extension
4. Evaluation
5. Summary
What is shellcode

- Shellcode: *exploit payload* that spawns a shell
What is shellcode

- **Shellcode**: exploit payload that spawns a shell
- ... or any other malicious code carried by an exploit
What is shellcode

Listing 1: Example Metasploit exploit [6]
Honeypots

- **honeypots** to encounter modern attacks
- systems without production value
- high- and low-interaction honeypots available
- direct interaction to **observe encrypted connections**
- major IPv6 general-purpose honeypots: Dionaea [3] and HoneydV6 [9]
- no shellcode detection support in HoneydV6 → **extend HoneydV6**
Why HoneydV6

- customised network stack in userspace
- simulate entire IPv6 networks with thousands of hosts
- dynamically creates virtual low-interaction honeypots
- monitor layer 3 attacks
Outline

1. Introduction
2. Shellcode detection and analysis
3. Honeypot shellcode detection extension
4. Evaluation
5. Summary
Shellcode detection and analysis

- identify traffic containing shellcode automatically
- analyse shellcode behaviour
- goal: **find and evaluate existing libraries** for HoneydV6 integration
Shellcode detection mechanisms

- pattern matching
Shellcode detection mechanisms

- pattern matching
- execution on a real OS
Shellcode detection mechanisms

- pattern matching
- execution on a real OS
- emulation
  - execute shellcode in a safe environment [8]
  - many papers but few implementations
  - libemu only open source library[2]
  - alternative Shellzer is limited to JS, Flash and PDF malware [4]
libemu

- C library developed in 2007
- used by Dionaea
- x86 emulator - registers, program counter, virtual memory, disassembler
- utilises address determination problem to locate code sequences
- `emu_shellcode_test()` returns position of detected shellcode sequence
- ability to trace accessed system calls
Online malware analysis

- Malwr [5]
  - web interface for Cuckoobox
- Anubis [1]
  - provides interface to upload shellcode samples
  - provides HTML/XML/PDF/ASCII result protocol
Integration of libemu and Anubis into HoneydV6

- added **shellcode buffer** to connection structures (*tcp_con*, *udp_con*)
- **extended callbacks** for traffic handling (*cmd_tcp_write*, *cmd_tcp_write*)
- **SQLite database** setup and connector
- background job uses libemu to mark and submit "interesting" received traffic
Modifications for Anubis

- support for Windows and Android binaries only
- msfencode to create unencrypted x86 binaries
- MD5 checksum generation for samples to avoid duplicates
- libcurl-based uploader for submission and report url logging
Outline

1. Introduction
2. Shellcode detection and analysis
3. Honeypot shellcode detection extension
4. Evaluation
5. Summary
Detection rate measurement setup

- Metasploit framework [6] to generate **107 shellcode samples**
- Dionaea with modified default configuration to accept **http requests**
- HoneydV6 configured with a single host running a web server
- Netcat [7] for shellcode transmission (different source ports for correlation)
- inspected both databases for traffic marked as malicious
Detection rate measurements results

- All shellcodes detected by Dionaea were also detected by HoneydV6.
- Both honeypots use libemu to detect shellcodes.
- Further malware profiling in Dionaea.
HoneydV6 shellcode buffer size variations

<table>
<thead>
<tr>
<th>Buffer Size</th>
<th>16</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256 - 8192</th>
</tr>
</thead>
<tbody>
<tr>
<td>#Detected samples</td>
<td>0</td>
<td>12</td>
<td>23</td>
<td>25</td>
<td>26</td>
</tr>
</tbody>
</table>

Table: HoneydV6 detection rate for different shellcode buffer sizes

- Measurements with **default buffer size of 1024 bytes**
- At least 31 bytes buffer needed to detect first sample
- Depending on exploit larger buffer sizes needed
Outline

1. Introduction

2. Shellcode detection and analysis

3. Honeypot shellcode detection extension

4. Evaluation

5. Summary
Summary

- IPv6 attack detection still in early stage
- integration of libemu into HoneydV6 is a first step
- only two general-purpose low-interaction honeypots available
- no further developed open source shellcode detection libraries available
Time for questions...
New HoneydV6 logging database

connection
- id
- source
- destination
- start_time
- end_time
- protocol
- protocol_id
- af_inet_type

has

tcp_udp
- id
- sport
- dport
- payload_id

contains

payload
- id
- state
- payload
- filename
- report_url

icmp
- id
- type
- code
References

   Anubis: Analyzing Unknown Binaries, nd.

   libemu – x86 Shellcode Emulation, nd.

   dionaea catches bugs.
   http://dionaea.carnivore.it/, nd.

   Shellzer: A tool for the dynamic analysis of malicious shellcode.

   Malwr - Malware Analysis by Cuckoo Sandbox, nd.

   Metasploit: Penetration Testing Software, nd.

   The GNU Netcat project, nd.

   Network level polymorphic shellcode detection using emulation.

   Ipv6 network attack detection with honeydv6.
   to appear.