Validation of the Network-based Dictionary Attack Detection

Jan Vykopal, Tomas Plesnik, Pavel Minarik
{vykopal, plesnik, minarik}@ics.muni.cz

Institute of Computer Science, Masaryk University
Brno, Czech Republic

SPI 2009, Brno
May 7th, 2009
Table of contents

1 Motivation
2 State of the art: Host-based approach
3 State of the art: Network-based approach
4 SSH dictionary attack pattern
5 Validation
6 Test bed
7 Results
8 Conclusions
root@medea: # grep sshd /var/log/auth.log | grep "Failed password"

Failed password for invalid user **test** from **208.53.158.47** port 45922 ssh2
Failed password for invalid user **rollie** from **208.53.158.47** port 46202 ssh2
Failed password for invalid user **oracle** from **208.53.158.47** port 46409 ssh2
Failed password for invalid user **boni** from **208.53.158.47** port 46666 ssh2
Failed password for invalid user **backup** from **208.53.158.47** port 46941 ssh2
Failed password for invalid user **fiedler** from **208.53.158.47** port 47185 ssh2
Failed password for invalid user **domain** from **208.53.158.47** port 47393 ssh2
Failed password for invalid user **web** from **208.53.158.47** port 54688 ssh2
Failed password for invalid user **client** from **208.53.158.47** port 54897 ssh2
Failed password for invalid user **clients** from **208.53.158.47** port 55169 ssh2
Failed password for invalid user **rose** from **208.53.158.47** port 55458 ssh2

...
Motivation II

- many break-in attempts (mainly by *script kiddies*)
- users are incorrigible $\Rightarrow$ use weak passwords
- administrators are lax

Main threats
- system resource abuse ($\Rightarrow$ insider attacks)
- identity theft
Host-based approach

Access log parsing:

Oct 1 19:34:11 medea sshd[17493]: Failed password for invalid user xyz from a.b.c.d port n ssh2

- DenyHosts\(^1\) and logwatch\(^2\) tools
- OS-specific tools

- not always deployable
- linear increase of maintenance effort

\(^1\)only for SSH servers, http://denyhosts.sourceforge.net/
\(^2\)http://www.logwatch.org/
Network-based approach

- almost constant maintenance effort
- transparent

- but no (free) usable tool or system known

- we want to detect attacks in the whole university network and have NetFlow data available ⇒ flow-based approach
Little excursion into the world of NetFlow

- an *unidirectional* sequence of packets with some common properties that pass through a network device
- flow is commonly 5-tuple \((IP_{src}, IP_{dst}, proto, srcPort, dstPort)\)
- not „interested“ in payload
- traffic aggregation (sum of packets, bytes, flow duration . . .)
- eligible for multigigabit networks
SSH dictionary attack in NetFlow data

Attacker A and victim B

<table>
<thead>
<tr>
<th>Flow start</th>
<th>dur</th>
<th>proto</th>
<th>src IP:port</th>
<th>dst IP:port</th>
<th>p</th>
<th>B</th>
<th>Bpp</th>
</tr>
</thead>
<tbody>
<tr>
<td>20:18:45.540</td>
<td>1.220</td>
<td>TCP</td>
<td>IP_A:45728</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:45.541</td>
<td>1.096</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:45728</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
<tr>
<td>20:18:46.636</td>
<td>1.210</td>
<td>TCP</td>
<td>IP_A:45824</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:46.636</td>
<td>1.088</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:45824</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
<tr>
<td>20:18:47.722</td>
<td>1.212</td>
<td>TCP</td>
<td>IP_A:45930</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:47.722</td>
<td>1.090</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:45930</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
<tr>
<td>20:18:49.922</td>
<td>1.247</td>
<td>TCP</td>
<td>IP_A:46104</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:49.922</td>
<td>1.121</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:46104</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
<tr>
<td>20:18:51.039</td>
<td>1.219</td>
<td>TCP</td>
<td>IP_A:46210</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:51.039</td>
<td>1.096</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:46210</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
<tr>
<td>20:18:52.132</td>
<td>1.216</td>
<td>TCP</td>
<td>IP_A:46317</td>
<td>IP_B:22</td>
<td>12</td>
<td>1020</td>
<td>85</td>
</tr>
<tr>
<td>20:18:52.133</td>
<td>1.092</td>
<td>TCP</td>
<td>IP_B:22</td>
<td>IP_A:46317</td>
<td>24</td>
<td>4274</td>
<td>178</td>
</tr>
</tbody>
</table>
SSH dictionary attack pattern

- focused on *TCP port 22* (= port-based classification)
- *many flows* from attacker to victim in *short time window*
- the flows are *small*\(^3\) (in terms of number and size of packets and bytes)
- responses are *small* too (typically the same number of packets and bytes)
- flow duration is *up to 2.5 seconds*\(^4\)
- in case of successful attempt the last flow is usually completely different

- pattern variables are in *italics*

\(^3\)from 20 to 30 small packets and from 2 000 to 5 000 bytes
\(^4\)in multigigabit networks
Implementation: adaptive decision tree

- Victim IP address
- Victim level
- Store: vAAC
- Duration level
- Packets level
- Attacker IP address, tsLA test
- Attacker level
- Store: aAAC, tsLA, Darr, Parr, Barr
- Flow duration out of bounds
- Flow duration in bounds
- Operations: SUCC
- Packets out of bounds
- Packets in bounds
- Operations: SUCC
- Bytes out of bounds
- Bytes in bounds
- Operations: SUCC
- Operations: Aup, Bup

J. Vykopal, T. Plesnik, P. Minarik vykopal@ics.muni.cz

Validation of the Network-based Dictionary Attack Detection
Validation

- previously known attacks
  - reliable detection of successful and unsuccessful “simple” and distributed attacks (against one victim)
  - newly detected attacks of previously known attackers
- no suitable data to estimate false negative rate

Goal of this paper: validate the detection pattern by honeypots
Test bed

Validation of the Network-based Dictionary Attack Detection
Honeypots

- 5 guests (Linux Ubuntu 8.10) in VMware Server 2.0
- patched OpenSSH 5.1p1 server
  - log with names and even **passwords**,
  - a copy of log stored in uncommon path,
  - e-mail alert sent after successful authentication via password.
- 10 user accounts with weak passwords on each guest
- any other services and daemons disabled
- all outbound traffic shaped to 32kbps by *Traffic Control*
NetFlow probes and collector

- FlowMon probe connected via Cisco SPAN (mirror) port
- another FlowMon probe monitors all traffic between guests and all other hosts
- NetFlow collector nfdump stores NetFlow records and serve NetFlow data to the detection module
Results

- **unsuccessful** attack: at least 20 repetitive attempts to log in
- one attack: at least two contiguous attempts in 30 minute

- totally 65 SSH dictionary attacks observed during 23 day
- only **3 attacks(!) were successful** (4.61%)
- less than 20 repetitive log-in attempts for 16 times
Results II

- 938 TCP and 501 UDP scans originating (only) outside the network
- the most popular TCP ports: 1433 (MS SQL) and 80 (HTTP), UDP ports: 1026 and 1027 (DCOM)
- 21 of 34 scans were followed by SSH dictionary attacks
- time between the scan and the attack: 6 minutes – 2 hours.
- only one successful attacks preceded by the common scan,
- other two successful attacks preceded by an establishing of TCP connection (SSH scan) to the port 22 (about 1 and 9 hours before the attack)
## Results III

<table>
<thead>
<tr>
<th></th>
<th>1st SA</th>
<th>2nd SA</th>
<th>3rd SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scan</td>
<td>SSH</td>
<td>TCP SYN/RST</td>
<td>SSH</td>
</tr>
<tr>
<td>Parallel attacks</td>
<td>5</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Logged by</td>
<td>guest/guest123</td>
<td>guest/12345</td>
<td>test/qwerty</td>
</tr>
<tr>
<td>Time</td>
<td>1m4s</td>
<td>3m6s</td>
<td>21m48s</td>
</tr>
<tr>
<td>Attempts</td>
<td>44</td>
<td>56</td>
<td>401</td>
</tr>
<tr>
<td>Pattern</td>
<td>username/username</td>
<td>–</td>
<td>username/username</td>
</tr>
<tr>
<td>Activity</td>
<td>none</td>
<td>none</td>
<td>none</td>
</tr>
<tr>
<td>A. cont.?</td>
<td>yes (2191 a.)</td>
<td>only scan</td>
<td>against other 3 hosts</td>
</tr>
</tbody>
</table>

J. Vykopal, T. Plesnik, P. Minarik  vykopal@ics.muni.cz

Validation of the Network-based Dictionary Attack Detection
Results IV

- all 14 (5+5+4) SSH dictionary attacks were detected
- no other attacks were detected
- attacks were correctly labeled as *successful* or *unsuccessful* except one
- attacker continued with the attack after successful intrusion

- performance: 466, 559 and 809 flows per second => sufficient for real campus network
Conclusions

- SSH dictionary attack pattern was validated and performance was measured
- sensitivity to NetFlow data quality (SPAN vs. tap) was confirmed
- behaviour of attackers was studied and compared to previous studies
Thank you for your attention. Questions?