Bruteforcing in the Shadows
Evading Automated Detection

Martin Drašar, Jan Vykopal

{drasar|vykopal}@ics.muni.cz

Institute of Computer Science
Masaryk University
Brno, Czech Republic

FloCon 2012
January 12, Austin, Texas

Part I

Network Security Monitoring at Masaryk University, Brno

Masaryk University and CSIRT-MU

Masaryk University, Brno

- **2nd largest** university in the Czech Republic.
- ~45,000 students, ~5,000 staff.
- ~**15,000 hosts** (public IPs) online every day.

CSIRT-MU

- In charge of university network security since 2009.
- Accredited by European Trusted Introducer in 2011.
- ~3 FTE ⇒ **automatization**.
- **Intrusion detection** is one of key services.

Intrusion Detection at Masaryk University

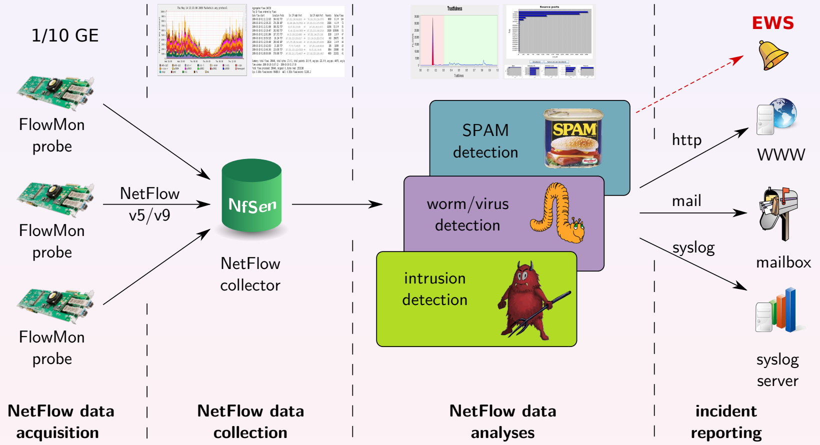
NetFlow-based

- ∼35 software (1 GE) or hardware accelerated (10 GE) stand-alone probes at important links.
- Several NfSen collectors for data storage.
- Wealth of in-house developed detection tools.

Other

- Low- and high-interaction honeypots.
- Integrated logging system based on Syslog.
- External data sources: Shadowserver and Team Cymru reports.

Intrusion Detection at Masaryk University contd.



In-house Developed Tools

Our tools detects:

- live hosts and missing DNS reverse entries,
- port scans,
- embedded botnets (presented at FloCon 2011),
- network address translators,
- spamming hosts,
- changes in RTT of selected servers,
- **bruteforce attacks.**

In progress: analytic tool of phishing incidents.

Part II

Bruteforcing: State of the Art

Bruteforce SSH Attacks: Introduction

- Repetitive online password guessing.
- Ubiquitous on the Internet.
- Humans tend to select week passwords.
- Still a threat (stepping stones, data leakage, ...).
- Common attack types – 1:1, 1:N, M:1, and M:N.

- Traditionally detected at the host level (1:1 and M:1).
- Host-based detection may miss large attacks (1:N and M:N).
- Network-based detection promises to protect devices that do not protect themselves (such as embedded devices, routers).

Diversion: Datasets & Data Sources

Detection methods are developed using various datasets, but academia uses data sets that do not reflect the reality.

- Datasets do not contain current attacks ⇒ **old**.
- Data are from limited topologies and small number of networked hosts ⇒ **small-scaled**.
- They are not (sufficiently) annotated, leave you on your own.
- Favor *volume-visible* attacks such as scans, floods, DoS, etc.
- May or may not contain stealth attacks.
- Security research bent on detecting attacks present in datasets only.

We prefer real examples from daily life of CSIRT-MU.

Bruteforce SSH Attacks: Detection

Typical brute force SSH attack scenario:

- Uses common SSH port.
- Repetitive password guessing generates *similar* flows in terms of volume and time.
- Small and short flows indicates unsuccessful attempts.
- It is often preceded by port scanning.
- One attacker aims at several (many) hosts.
- Attacks continue even after black-holing.

We believe it is the similar case for other services/protocols: Telnet, RDP, web authentication, etc.

Increasing Effectiveness of Detection

- Know your network. Is it server or client?
- Have the attackers scanned our network already?
- Do they access honeypots?
- Are they from untrusted autonomous system?
- Use external data sources but be careful!

Part III

Evading Automated Detection

Black-box Probing

Typical bruteforce attack [THC-Hydra, ...]

- Attacking as much as it is possible.
- Using predefined attacks per time frame.

Sophisticated bruteforce attack [Ncrack]

- Adaptive lowering of attack attempts per time frame to get under network thresholds.
- Threshold detection is not that hard.
  - Especially when there is an imminent blocking of attackers.
- Possible if attacker controls more hosts.

Random Delays

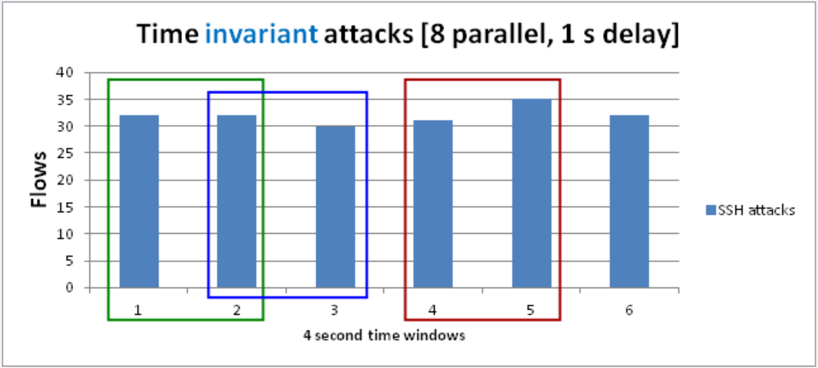
Typical bruteforce attack [THC-Hydra, . . .]

- Fixed attempts per time frame.
- Zero or fixed interval between attempts.

Sophisticated bruteforce attack [Ncrack]

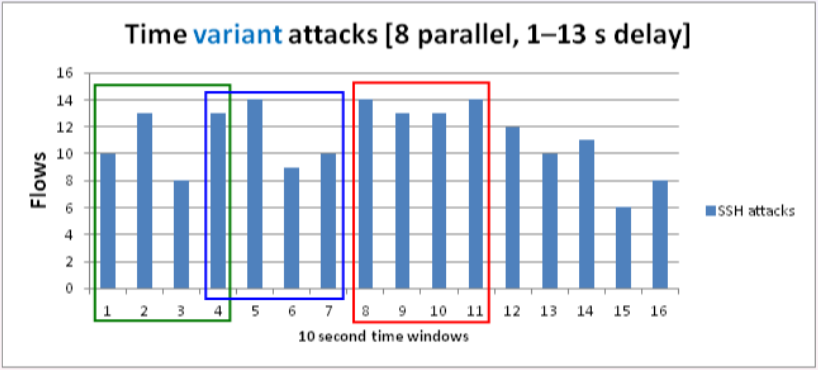
- Simulation of real traffic by inserting random delays between attack attempts.
- Random delays cause variations in the number of attempts per timeframe.

Random Delays: Illustration



green ~ blue ~ red

Random Delays: Illustration contd.



green !~ blue !~ red

Flow Stretching

Typical bruteforce attack [THC-Hydra, Ncrack, . . .]

- Exchanging bare minimum of data needed for authentication attempt.

Sophisticated bruteforce attack [?]

- Flow-wise the volume and duration are the only difference between successful and failed authentication.
- Exploiting protocol design to mimic successful authentication by increasing volume and duration.
- SSH, RDP, HTTP and probably others.

Flow Stretching: Illustration

Non-stretched flows

Duration	Protocol	Src IP:Src Port		Dst IP:Port	Packets	Bytes
1.310	TCP	147.251.AA.BB:49297	->	147.251.CC.DD:22	12	1197
0.269	TCP	147.251.AA.BB:49320	->	147.251.CC.DD:22	11	1157
0.436	TCP	147.251.AA.BB:49329	->	147.251.CC.DD:22	11	1157
0.196	TCP	147.251.AA.BB:49358	->	147.251.CC.DD:22	11	1173
0.155	TCP	147.251.AA.BB:49308	->	147.251.CC.DD:22	11	1157
0.273	TCP	147.251.AA.BB:49318	->	147.251.CC.DD:22	11	1157
0.270	TCP	147.251.AA.BB:49343	->	147.251.CC.DD:22	11	1157
0.259	TCP	147.251.AA.BB:49344	->	147.251.CC.DD:22	11	1157
0.206	TCP	147.251.AA.BB:49355	->	147.251.CC.DD:22	11	1173
0.190	TCP	147.251.AA.BB:49362	->	147.251.CC.DD:22	11	1157

Flow Stretching: Illustration contd.

Stretched flows

Duration	Protocol	Src IP:Src Port		Dst IP:Port	Packets	Bytes
8.157	TCP	147.251.AA.BB:49368	->	147.251.CC.DD:22	142	44441
5.501	TCP	147.251.AA.BB:49379	->	147.251.CC.DD:22	99	30389
14.227	TCP	147.251.AA.BB:49367	->	147.251.CC.DD:22	239	76837
6.722	TCP	147.251.AA.BB:49369	->	147.251.CC.DD:22	119	36981
5.429	TCP	147.251.AA.BB:49372	->	147.251.CC.DD:22	98	29865
18.184	TCP	147.251.AA.BB:49375	->	147.251.CC.DD:22	302	97593
2.239	TCP	147.251.AA.BB:49387	->	147.251.CC.DD:22	47	13125
1.304	TCP	147.251.AA.BB:49380	->	147.251.CC.DD:22	32	8033
23.320	TCP	147.251.AA.BB:49374	->	147.251.CC.DD:22	384	124865
1.798	TCP	147.251.AA.BB:49386	->	147.251.CC.DD:22	40	10737

Search for a Remedy

- Correlation with (sys)log information?
- Bigger restrictiveness?
- More sophisticated flow analyses?
- **Open problem...**

Summary

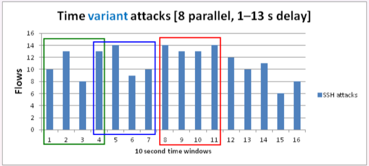
- Brute-force (SSH) attacks are still a threat.
- It is possible to detect them by NetFlow analysis.
- Defenders are prepared for
  simple attacks and simple attackers.
- Many NetFlow detection methods can be easily evaded.
- Limitations of NetFlow – how to overcome them?

Bruteforcing in the Shadows
Evading Automated Detection

Martin Drašar
Jan Vykopal

{drasar|vykopal}@ics.muni.cz

Project CYBER
http://www.muni.cz/ics/cyber



green !~ blue !~ red

This material is based upon work supported by the
Czech Ministry of Defence under Contract No. OVMASUN200801.