Masterarbeits

Structure embeddings for OpenSSH heap dump analysis

A report by

Lahoche, Clément Claude Martial

Prüfer

Prof. Dr. Michael Granitzer

Christofer Fellicious

Prof. Dr. Pierre-Edouard Portier

August 23, 2023

Abstract

Acknowledgements

Contents

1 Introduction **1**

2 Research Questions **1**

3 Structure of the Thesis **1**

4 Background **2**

4.1 Graph Generation from Heap Dumps 3

4.1.1 Introduction to heap dumps in OpenSSH 3

4.1.2 Definitions : Structures, Pointers, and the role of malloc headers 3

4.2 Traditional Statistical Embedding 3

4.2.1 Shannon entropy and its application in byte sequence analysis 3

4.2.2 Byte Frequency Distribution (BFD) and its significance 3

4.2.3 Bigram, trigram frequencies 3

4.3 Deep Learning Models for Raw Byte Embedding 3

4.3.1 Introduction to the role of deep learning in byte sequence analysis 3

4.3.2 RNNs : Understanding sequence data 3

4.3.3 CNNs : Pattern detection in raw bytes 3

4.3.4 Autoencoders 3

4.3.5 Transformers 3

4.4 Graph Embedding Methods 3

4.4.1 Introduction to graph embedding 3

4.4.2 Popular embedding techniques 3

4.4.3 Applications and significance in OpenSSH heap dump analysis 3

4.5 Conclusion and Transition to the Next Section 3

5 Methods **4**

6 Results **5**

7 Discussion **6**

8 Conclusion **7**

Appendix A Code **8**

Appendix B Math **8**

Appendix C Dataset **8**

Acronymes **9**

References **10**

Additional bibliography **10**

List of Figures

List of Tables

1 Introduction

Digital forensics is a linchpin in cybersecurity, enabling the extraction of vital evidence from devices like PCs. This evidence is key for detecting malware and tracing intruder activities. Analyzing a device's main memory is a go-to technique in this field. The fusion of machine learning promises to amplify and streamline these analyses.

With the rising need for encrypted communication, Secure Shell (SSH) protocols are now commonplace. However, these security-focused channels can inadvertently shield malicious actions, posing challenges to standard investigative approaches. Cutting-edge research offers solutions. The work in *SmartKex: Machine Learning Assisted SSH Keys Extraction From The Heap Dump* [3] highlights how machine learning can boost the extraction of session keys from OpenSSH memory images. In a complementary vein, „SSHkex: Leveraging virtual machine introspection for extracting SSH keys and decrypting SSH network traffic“ [7] showcases the power of Virtual Machine Introspection (VMI) for direct SSH key extraction.

Inspired by *SmartKex: Machine Learning Assisted SSH Keys Extraction From The Heap Dump*, this thesis zeroes in on a central challenge: data embedding. While previous studies set the stage for key extraction, the data embedding technique, especially windowing, can be optimized. The design of data embeddings is pivotal for machine learning efficacy, especially in nuanced tasks like memory analysis. This research introduces fresh embedding strategies, aiming to refine extraction and unearth deeper memory snapshot patterns. Merging graph embeddings with advanced machine learning, the goal is to craft a sophisticated toolkit for OpenSSH heap dump studies, bridging digital forensics and machine learning.

2 Research Questions

Write down and explain your research questions (2-5)

3 Structure of the Thesis

Explain the structure of the thesis.

4 Background

In the complex world of cybersecurity and digital forensics, innovative approaches are crucial for revealing hidden or encrypted information. OpenSSH stands out as a key instrument for ensuring secure communication. The memory snapshots, or heap dumps, of OpenSSH are treasure troves of data. Through graph generation from these dumps, we can uncover the detailed connections between data structures, identified by their malloc headers, and their associated pointers.

This research delves deep into the smart embedding of these connections, aiming to use machine learning classifiers to identify structures that contain OpenSSH keys. The journey is not just about representing data through graphs but also about understanding the raw sequences of bytes in the heap dump. Classical techniques like Shannon entropy, Byte Frequency Distribution (BFD), and bigram frequencies provide foundational knowledge. However, the rapidly evolving domain of deep learning opens up a plethora of avenues. Models such as Recurrent Neural Networks (RNNs) [6] (like Long Short-Term Memory (LTSM) [5] and Gated Recurrent Units (GRU)[2]) and sequence-to-sequence learning [8] offer unique perspectives on raw byte embedding. The transformative power of attention mechanisms, as highlighted by the transformer architecture[9]. Furthermore, the efficacy of convolutional approaches (CNN), both standalone and in conjunction with recurrent networks, for sequence modeling is well-documented [1]. Notably, the application of neural networks in file fragment classification, especially with lossless representations, has shown promising results [4].

The aim of this background section is to provide a comprehensive overview of graph creation from heap dumps, techniques for raw byte embedding, and their role in identifying OpenSSH key structures. By merging age-old techniques with modern approaches, we strive to highlight the most effective methods for analyzing OpenSSH heap dump.

4.1 Graph Generation from Heap Dumps

4.1.1 Introduction to heap dumps in OpenSSH

4.1.2 Definitions : Structures, Pointers, and the role of malloc headers

4.2 Traditional Statistical Embedding

4.2.1 Shannon entropy and its application in byte sequence analysis

4.2.2 Byte Frequency Distribution (BFD) and its significance

4.2.3 Bigram, trigram frequencies

4.3 Deep Learning Models for Raw Byte Embedding

4.3.1 Introduction to the role of deep learning in byte sequence analysis

4.3.2 RNNs : Understanding sequence data

4.3.3 CNNs : Pattern detection in raw bytes

4.3.4 Autoencoders

4.3.5 Transformers

4.4 Graph Embedding Methods

4.4.1 Introduction to graph embedding

4.4.2 Popular embedding techniques

Node2Vec, GraphSAGE, and others

4.4.3 Applications and significance in OpenSSH heap dump analysis

4.5 Conclusion and Transition to the Next Section

5 Methods

Describe the method/software/tool/algorithm you have developed here

6 Results

Describe the experimental setup, the used datasets/parameters and the experimental results achieved

7 Discussion

Discuss the results. What is the outcome of your experimetns?

8 Conclusion

Summarize the thesis and provide a outlook on future work.

A Code

B Math

C Dataset

Acronymes

BFD Byte Frequency Distribution. 2

CNN Convolutional Neural Networks. 2

GRU Gated Recurrent Units. 2

LTSM Long Short-Term Memory. 2

RNN Recurrent Neural Networks. 2

SSH Secure Shell. 1

VMI Virtual Machine Introspection. 1

References

[1] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. Apr. 19, 2018. arXiv: 1803.01271[cs]. URL: http://arxiv.org/abs/1803.01271 (visited on 08/23/2023).

[2] Junyoung Chung et al. *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*. Dec. 11, 2014. arXiv: 1412.3555[cs]. URL: http://arxiv.org/abs/1412.3555 (visited on 08/23/2023).

[3] Christofer Fellicious et al. *SmartKex: Machine Learning Assisted SSH Keys Extraction From The Heap Dump*. Sept. 13, 2022. arXiv: 2209.05243[cs]. URL: http://arxiv.org/abs/2209.05243 (visited on 08/17/2023).

[4] Luke Hiester. „File Fragment Classification Using Neural Networks with Lossless Representations“. In: *East Tennessee State University* (May 2018). (Visited on 08/21/2023).

[5] Sepp Hochreiter and Jürgen Schmidhuber. „Long short-term memory“. In: *Neural computation* 9.8 (1997). Publisher: MIT Press, pp. 1735–1780. (Visited on 08/23/2023).

[6] Siwei Lai et al. „Recurrent Convolutional Neural Networks for Text Classification“. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 29.1 (Feb. 19, 2015). ISSN: 2374-3468, 2159-5399. DOI: 10.1609/aaai.v29i1.9513. URL: https://ojs.aaai.org/index.php/AAAI/article/view/9513 (visited on 08/23/2023).

[7] Stewart Sentanoe and Hans P. Reiser. „SSHkex: Leveraging virtual machine introspection for extracting SSH keys and decrypting SSH network traffic“. In: *Forensic Science International: Digital Investigation* 40 (Apr. 2022), p. 301337. ISSN: 26662817. DOI: 10.1016/j.fsidi.2022.301337. URL: https://linkinghub.elsevier.com/retrieve/pii/S2666281722000063 (visited on 08/17/2023).

[8] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. *Sequence to Sequence Learning with Neural Networks*. Dec. 14, 2014. arXiv: 1409.3215[cs]. URL: http://arxiv.org/abs/1409.3215 (visited on 08/23/2023).

[9] Ashish Vaswani et al. „Attention Is All You Need“. In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 5998–6008. (Visited on 08/23/2023).

Additional bibliography

[10] Vivek Gite. *How To Reuse SSH Connection To Speed Up Remote Login Process Using Multiplexing*. nixCraft. Aug. 20, 2008. URL: https://www.cyberciti.biz/faq/linux-unix-reuse-openssh-connection/ (visited on 10/21/2022).

[11] Weijie Huang and Jun Wang. *Character-level Convolutional Network for Text Classification Applied to Chinese Corpus*. Nov. 15, 2016. arXiv: 1611.04358[cs]. URL: http://arxiv.org/abs/1611.04358 (visited on 08/17/2023).

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich diese Masterarbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe und alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, als solche gekennzeichnet sind, sowie, dass ich die Masterarbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt habe.

Passau, August 23, 2023

Lahoche, Clément Claude Martial