www.crysys.hu

Network Security (BMEVIHIMB00)

Practical Session: Sandboxes and Honeypots

Gergő Ládi
Laboratory of Cryptography and System Security
Department of Networked Systems and Services
Gergo.Ladi@CrySyS.hu

© 2023 Laboratory of Cryptography and System Security, Budapest University of Technology and Economics

Outline

- Sandboxes
  - What is a sandbox?
  - Use cases and considerations
  - Demo
- Honeypots
  - What is a honeypot?
  - Different kinds of honeypots
  - Use cases and considerations
  - Demo
- This presentation is based on Tamás Holczer's slides

SANDBOXES

Sandboxing

- The aim is to run executables in a controlled environment
- Controlled environment: sandbox
  - Controlled access to files and other local resources
  - Controlled access to the network
  - It may contain emulated resources (e.g. fake devices)
  - It may be possible to take snapshots and return to them
  - It may be easier to take screenshots, collect memory dumps
- The sandbox is typically a separate virtual machine, but some software solutions exist that support running software sandboxed on a host OS

Sandboxing

- Why sandbox?
  - Malware analysis
  - Isolating known vulnerable applications that must be kept running
  - Protection against potentially misbehaving (buggy) software
- Examples
  - Sandboxie
  - Bochs
  - Cuckoo
  - Any.run
  - Virustotal
  - Joe Sandbox
  - Virtualization/containerization (ESXi, Hyper-V, Docker, …)
  - "Sacrificial lamb"

Sandboxing – Considerations

- Security considerations – sandboxing has its own dangers
  - Guest-to-host attacks (if using VMs)
  - Clipboard sharing
  - File sharing
- Sandbox detection – software might detect that they are being sandboxed (and behave differently)
  - Hardware IDs, names
  - Running services
  - Presence of integration tools
  - Other artifacts (hooks, timing, CPU behaviour, …)
- Some software need access to hardware (e.g. a microphone)
  - Can we make that work?
  - Can we make that work without compromising security?

Sandboxing – Demo

- VirusTotal
- Any.run
- Joe Sandbox

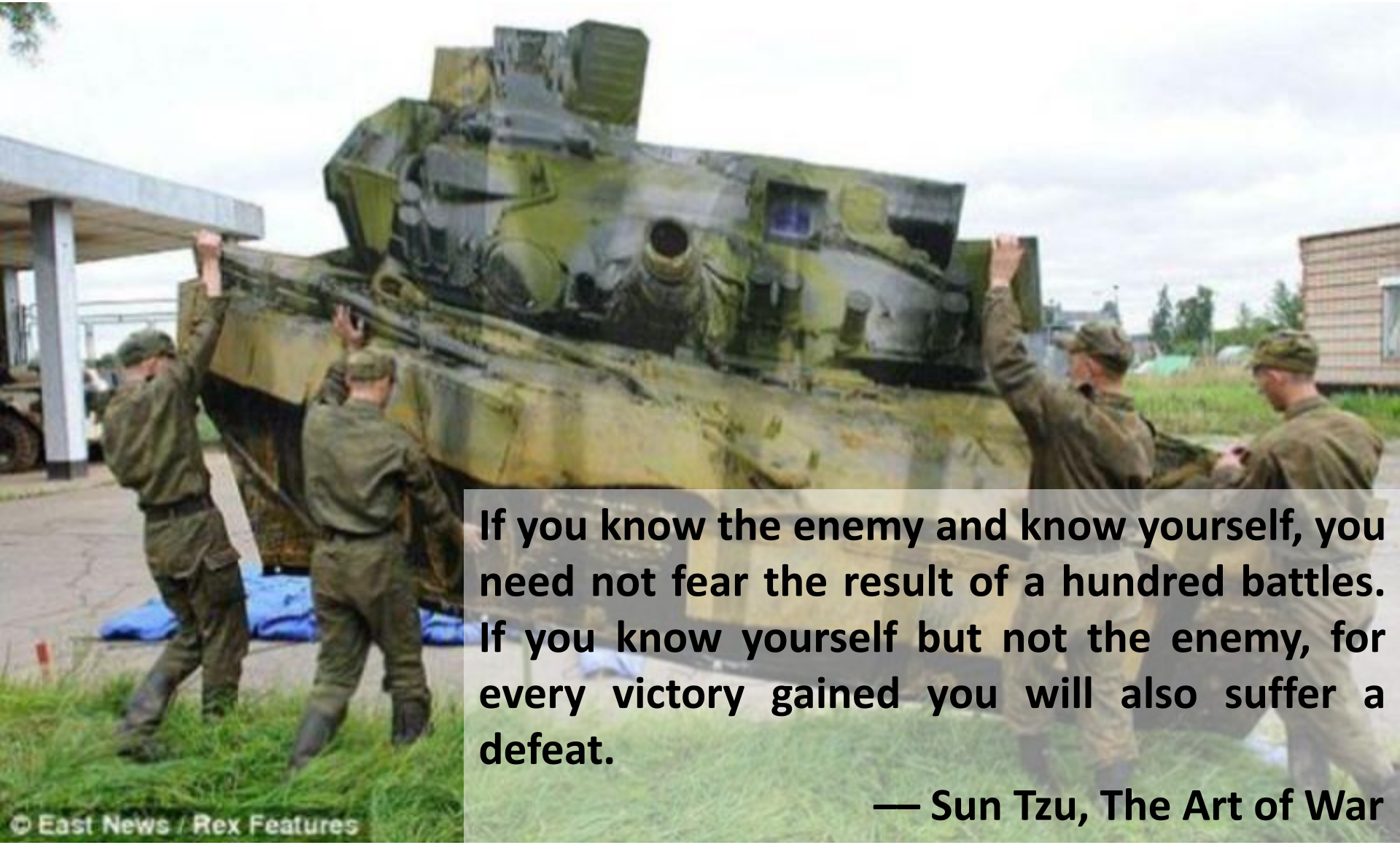
- Sandboxie

HONEYPOTS

Honeypots: The art of deception

Honeypots: The art of deception

Honeypots: The art of deception



If you know the enemy and know yourself, you need not fear the result of a hundred battles. If you know yourself but not the enemy, for every victory gained you will also suffer a defeat.

— Sun Tzu, The Art of War

Honeypots

- Definitions (L. Spitzner, 2002)
  - A security resource whose value lies in being probed, attacked, or compromised
  - An information system resource whose value lies in unauthorized or illicit use of that resource
- Any traffic to or from a honeypot is likely to be malicious
- Honeypots have no inherent "production" value, but may prove to be a valuable addition to a network
  - Early threat detection
  - Possible detection of previously unkown attacks and attack methods
  - Waste of time for the attacker (== more time for the defenders)
  - Logs and other data for analysis
  - Easy to restore a known good state after an incident

Honeypots – Classification

- By the level of interaction
  - High
  - Low
  - (Something in between?)
- By purpose
  - Production
  - Research
- By the implementation
  - Virtual
  - Physical

Honeypots – Levels of interaction

- Low interaction
  - Simulates only some aspects of the system
  - Easy to deploy
  - Minimal risk of compromise
  - Provides limited information about attacks and attackers
  - Example: Honeyd

- High interaction
  - Simulates all aspects of the OS, usually real systems
  - More difficult to deploy
  - Higher risk of compromise
  - Provides more information about attacks and attackers
  - Example: Honeynet

Honeypots – Implementation

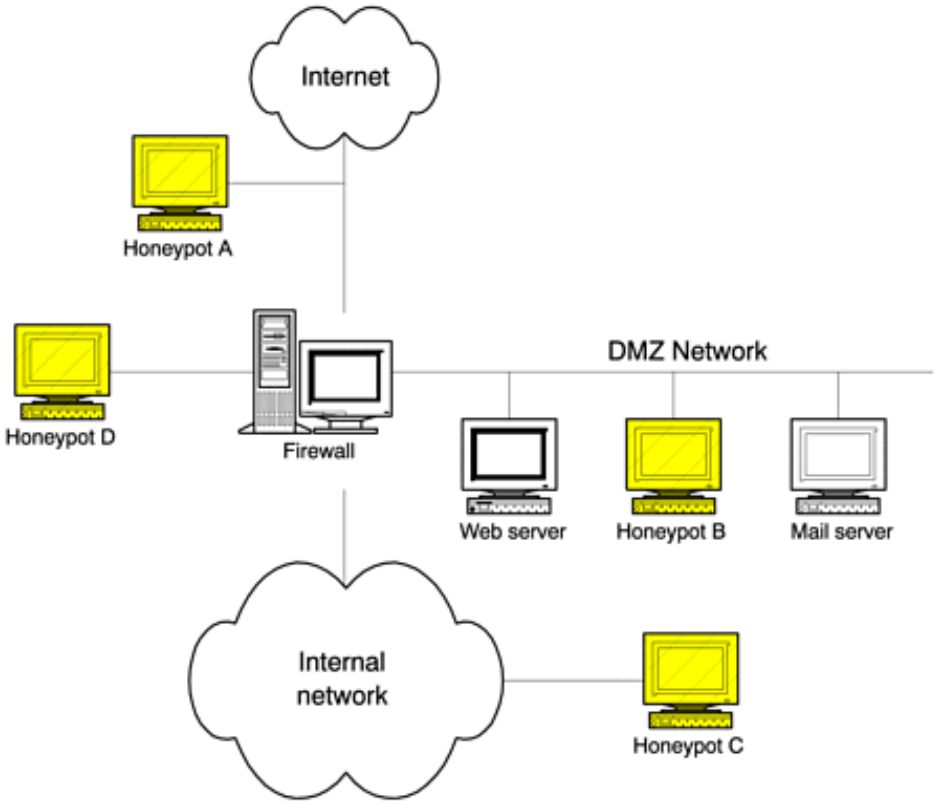
- Physical
  - Running on real, phyiscal machines
  - Each machine has its own IP address
  - Usually used in high-interaction scenarios
  - Least chance of getting detected as a honeypot by the attackers
- Virtual
  - Virtualizing physical honeypots (1->1)
    - » Easier to manage (snapshots, memory dumps, …)
    - » More likely to be detected as a honeypot
  - Simulated systems
    - » One system acts as if it was several others (many services on many IPs, …)

Honeypots – Purpose

- Production environments
  - Prevention
    - » Not effective for prevention, but may win the defenders time
    - » Ineffective against untargeted and automated attacks (e.g. worms, auto-rooters)
  - Detection
    - » Excellent early warning solution
    - » Another source of information to the SIEM
  - Response
    - » Easy to pull offline in case of an incident
    - » Useful data for forensics
- Research, education
  - High amounts of high value information if operated correctly
  - May lead to the discovery of new attacks, tools, vulnerabilities, tactics, …
  - Helps understand motives, behaviour, and organization
  - Development of analysis and forensic skills

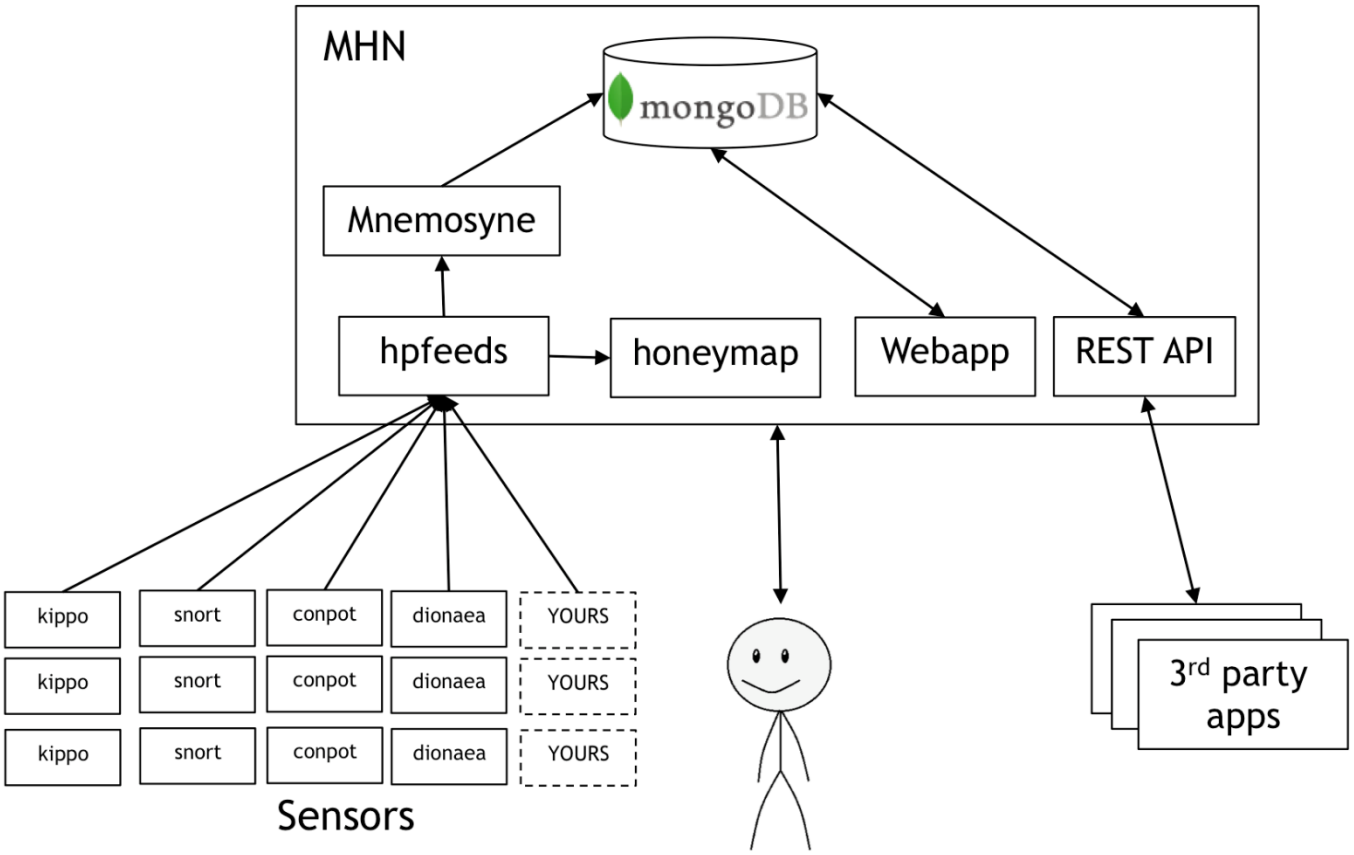
Honeypots – Placement

- What's the best place for a honeypot?
  - Practically anywhere!
- Typical locations
  - On the Internet (in front of the firewall)
  - In the demilitarized zone
  - Behind the firewall (inside the internal network)
- Different locations provide different information about different attacks and attackers
  - It might be a good strategy to have one in every segment of the network



Honeypots – Demo

- MHN: Modern Honey Network
  - https://github.com/pwnlandia/mhn



MISCELLANEOUS

Control Questions

- What is a sandbox?
- What are sandboxes used for?
- What should you consider when using a sandbox?

- What is a honeypot?
- What are the benefits of using honeypots?
- What is the difference between low-interaction and high-interaction honeypots?
- Where should honeypots be installed in a network? Why?

www.crysys.hu

Thank you for your attention!

That's all, folks!

Gergő Ládi
Laboratory of Cryptography and System Security
Department of Networked Systems and Services
Gergo.Ladi@CrySyS.hu

© 2023 Laboratory of Cryptography and System Security, Budapest University of Technology and Economics