Industrial Honeypots: know your enemy

Jaime Souto Casares

Inception and Definition

In 1986, during the last verses of the Cold War, Clifford Stoll, an astronomer turned sysadmin working in the Lawrence Berkeley National Laboratory, exposed and discovered the hacking activities of Markus Hess, a West German student who was selling information to the Soviet intelligentsia. After learning about the unauthorized access, Stoll and his then-girlfriend Martha Matthews came up with a plan to catch the intruder: to leave a load of dull government documents and directives, slightly modified to look like military classified data, and named *SDInet* (after Reagan's Strategic Defense Initiative), enticing enough to catch the hacker's attention and to keep them busy while the connection was being traced. The trap worked, leading to Hess' arrest by the German authorities and trial in 1990.[1,2]

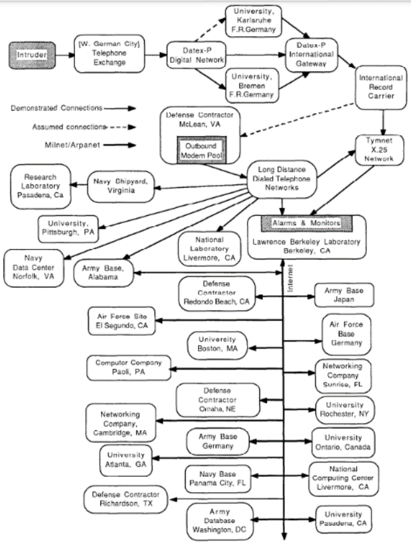
Intruder
[W. German City] Telephone Exchange
Datex-P Digital Network
University, Karlsruhe F.R.Germany
University, Bremen F.R.Germany
Datex-P International Gateway
International Record Carrier
Demonstrated Connections
Assumed connections
Milnet/Arpanet
Defense Contractor McLean, VA
Outbound Modem Pool
Tymnet X.25 Network
Long Distance Dialed Telephone Networks
Research Laboratory Pasadena, Ca
Navy Shipyard, Virginia
University, Pittsburgh, PA
Navy Data Center Norfolk, VA
Army Base, Alabama
National Laboratory Livermore, CA
Alarms & Monitors
Lawrence Berkeley Laboratory Berkeley, CA
Internet
Defense Contractor Redondo Beach, CA
Army Base Japan
Air Force Site El Segundo, CA
Air Force Base Germany
University Boston, MA
Computer Company Paoli, PA
Networking Company Sunrise, FL
Defense Contractor Omaha, NE
University Rochester, NY
Networking Company, Cambridge, MA
Army Base Germany
University Ontario, Canada
University Atlanta, GA
Navy Base Panama City, FL
National Computing Center Livermore, CA
Defense Contractor Richardson, TX
Army Database Washington, DC
University Pasadena, CA

Figure 1: Part of the notes taken by Clifford Stoll, following the attacker steps (taken from Ref. 2).

This was the first known occurrence of a honeypot, a cybersecurity device set up purposefully vulnerable to either lure attackers in or to serve as a decoy. Nowadays a mature and classic technique, honeypots are also organized in networks, the so-called honeynets, to better simulate a production environment.[3]

Honeypot Classification

Honeypots can be classified following different metrics.

By purpose: as aforementioned, a honeypot can have two main motivations. We can talk about a research honeypot: by letting the attackers in, a honeypot administrator can learn about the techniques used by nefarious users, and monitor their patterns in order to better enforce the security of the legitimate system. There are also production honeypots, that divert the attention of a hacker from the actual critical network, letting them think that they have actually managed to access.

By interactivity: depending on the level of "realness" of the system contained in the honeypot, one can classify them by interaction levels. High-interaction honeypots will mimic the structure and services of an actual system, with apparently legit data and functions within. They are rather expensive to set up in terms of resources, but attackers will have a much harder time detecting that they are dealing with a fake system. On the other side of the interactivity spectrum we have low-interaction honeypots, where only the basic structure of a generic operating system will be placed. Much simpler and easier to maintain, but trespassers will quickly reckon that they are in a honeypot (and might use that information to retaliate).

By role: another classification that interlocks with interactivity is the role of the honeypot. The ones described before are called server honeypots, passive entities waiting for attackers, but one could also set up a client honeypot, an active service like a web browser intended to find, interact with, and learn about malicious or unsecure servers or hosts.

Other types of interesting honeypots are database honeypots, false tables vulnerable to SQL injection, email honeypots used to attract and collect spam, and spider honeypots, prepared to detect bots like web crawlers.

Honeypots in OT systems

As in pretty much every aspect of cybersecurity, honeypots were first used in Information Technology (IT) systems. However, the ever-growing pressure that Operational Technology (OT) systems are experiencing since 2010, when the Stuxnet worm infected an Iranian nuclear facility, is read today as a call-to-action to fortify industrial systems against cyberattacks. But since the realms of IT and OT are very different, those techniques used in the former need to be revisited in order to be of any use for the latter, and honeypots are no exception. In particular, industrial systems ought to be extremely resilient: any impact on a critical infrastructure of industrial nature, such as energy production or water treatment facility, can have a huge impact on the physical world and on the wellbeing of entire human settlements or other ecosystems. This means that extra care has to be placed in a correct segmentation of the network to prevent potential leaps of the attacker from the honeynet to the actual system that we need to protect. Moreover, the structure of OT networks tends to be less structured and less predictable, which leads to a higher workload in order to simulate a quality high-interaction honeypot that can deceive attackers. This can be a really exhausting task, since some search engines like Shodan[4] greatly simplify the task of discovering whether a service is indeed a honeypot, unveiling the trick. Some researchers even reported their surprise when they noticed the large number of OT systems that were unnecessarily connected to the Internet, often with deficient security implementations.[5]

As explained in Ref. 6, we can talk about three generations of honeynets. *Generation I* (1999) was composed of a firewall and an Intrusion Detection System (IDS), with single honeypots behind. Although it worked well as a log collector, it was very transparent for the occasional intruder. *Generation II* (2002) improved the gateway, that now features an IDS sensor and an inline firewall operating as an invisible layer-two bridge. Because of that, there are no time-to-live decrements or MAC addresses for the intruder to play with.[7] Another capability is the ability to capture data from the attacker, sniffing their packets. Note that before these developments, a honeypot was not secure enough to be implemented in OT systems. *Generation III* (2004) included several deployment and management improvements, deepening on the data capture mechanisms.[8]

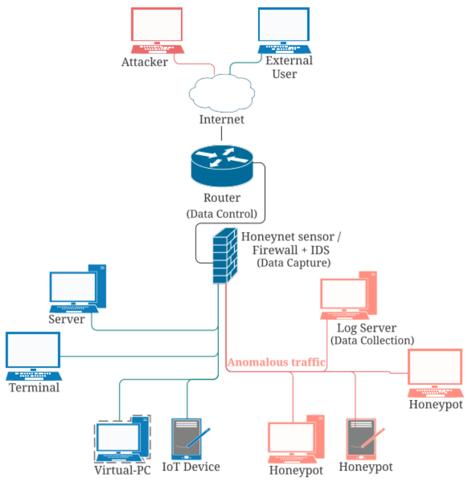

Figure 2: Basic OT honeynet architecture (taken from Ref. 9).

Following, we list several projects that have tackled these problems, each with their own strategy and tailored for different systems.

·Conpot:[10] one of the most popular OT honeynets, it is a low-interaction and open-source, hence easy to deploy and customize. Supports the main OT protocols (Modbus, S7comm) and also other protocols such as HTTP or FTP. Integrated under the Honeynet Project

·GasPot:[11] Also open-source, it was designed to mimic a Guardian AST gas-tank-monitoring system.

·Gridpot:[12] Open-source project that simulated an SCADA from an electric grid.

In 2016, Piggin and Buffey shared a detailed report[13] on an implementation of a high-interaction OT honeypot developed in 2014, with details about the actual construction of the system, its components and how to make it attractive to intruders. Interestingly, they listed the types of attacks registred by the honeypot:
- A password attack using default vendor credentials against an SCADA.
- An attempt to execute malicious code.
- Dictionary attacks.
- Brute force attacks over ssh.
- A focused attack against a PLC.
- An attempt to disrupt PLC data communications.
Unsurprisingly, the bulk of the attacks were automated reconnaissance scanning.
USA, China and the UK were, by this order, the greatest source of connections, and HTTP was reported as the most common protocol, followed by Remote Desktop Protocol. The authors also highlighted the extra steps taken by the attackers to hinder the localization of their connections, via proxies, VPN and TOR networks.

They repeated the experiment in 2018,[14] and their findings perfectly reflect the known trend of increasing risk over OT systems. They found a one-hundred fold growth in the activity, and that the interactions involving the Modbus protocol, now exposed in this study, were clearly intentional. Attacks came from all over the world, with only a portion of them easily attributed to researchers and universities. They did not find, however, any attacker reading or writing process data.

Summary

In this short overview, we gave the general characteristics of a honeypot: what they are, how to classify them, why they are relevant. Also, we talk about the importance of honeypots and honeynets in the realm of industrial networks.

References

[1] C. Stoll, *The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage* (1989).

[2] C. Stoll, *Stalking the Wily Hacker*, Commun. ACM 31, 484 (1988).

[3] L. Spitzner, *The Honeynet Project*, `https://www.honeynet.org`

[4] `https://www.shodan.io/`

[5] K. Wilhoit and S. Hilt*, The GasPot Experiment: Unexamined Perils in Using Gas-Tank-Monitoring Systems*, Trend Micro Incorporated, 201.

[6] A. D. Oza et al., *Survey of Snaring Cyber Attacks on IoT Devices with Honeypots and Honeynets,* 3rd International Conference for Convergence in Technology (2018).

[7] L. Spitzner, *The Honeynet Project: Trapping the Hackers,* IEEE Security & Privacy Magazine 1, 15 (2003).

[8] R. Siles, `https://community.broadcom.com/symantecenterprise/viewdocument/sebek-3-tracking-the-attackers-pa`

[9] J. Franco et al., *A survey of honeypots and honeynets for internet of things, industrial internet of things, and cyber-physical systems,* IEEE Communications Surveys & Tutorials, 23(4), 2351 (2021).

[10] `http://conpot.org/`

[11] `https://github.com/sjhilt/GasPot`

[12] `https://github.com/sk4ld/gridpot`

[13] R. Piggin and I. Buffey, *Active defence using an operational technology honeypot*, 11th International Conference on System Safety and Cyber-Security, pp. 1-6 (2016).

[14] https://www.snclavalin.com/en/beyond-engineering/the-ot-honeypot-reloaded