SSH attack on different honeypots

Erika Kovács

Table of contents

01 Introduction

Introduction

Why honeypots?

For the first time in my life, I heard about honeypots in a series.

What was the goal?

To explore a subject that we haven't covered in class yet.

02 Honeypots

What is a honeypot?

Quote

“Keep your friends close and your enemies closer.”

Sun Tzu

Behaviour

They enable a detailed study of cyber attack tactics, enhancing professionals' understanding of attacker methodologies.

Value

They play a crucial role in generating threat intelligence by collecting data on emerging threats and attack patterns.

Type of Honeypots

Web
WordPress login, Tomcat

Database
Elasticsearch, MySQL

Exploit
Citrix, Log4Pot

SSH
Endlessh, Cowrie

SMTP
Shiva

...and so on

Why SSH honeypots?

- Common Target
- Realism in Attacks
- Collection of Authentication Information
- Simulation of Cyber Attacks

My selections

Endlessh

"SSH tarpit that very slowly sends an endless, random SSH banner"

https://github.com/skeeto/endlessh

SSH-auth-logger

"Logs all authentication attempts as json making it easy to consume in other tools."

https://github.com/JustinAzoff/ssh-auth-logger

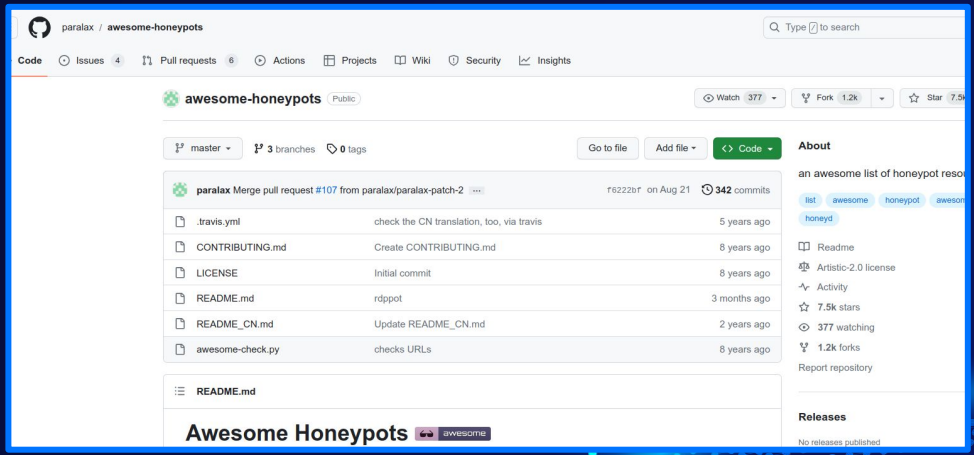
Cowrie

"Log brute force attacks and the shell interaction performed by the attacker. "

https://github.com/cowrie/cowrie

Source

https://github.com/paralax/awesome-honeypots



Comparison

	Endlessh	SSH-auth-logger	Cowrie
Programming language	C	Golang	Python
Mechanism	Slowly sends a large random SSH banner	Always rejects the login attempts	Provides a fake shell to the attacker if it logs in successfully
Useful against	Bots	Brute force	Human and bots
Logs	Attackers IP address only	Used credentials and IP addresses	Used credentials and the commands in the fake shell
Usage	To lure the attacker from real SSH port	For checking leaked employee usernames/passwords	Ways to allow/deny different credentials
Systemd service	yes	yes	yes

03 Systemd services

Managing with systemd

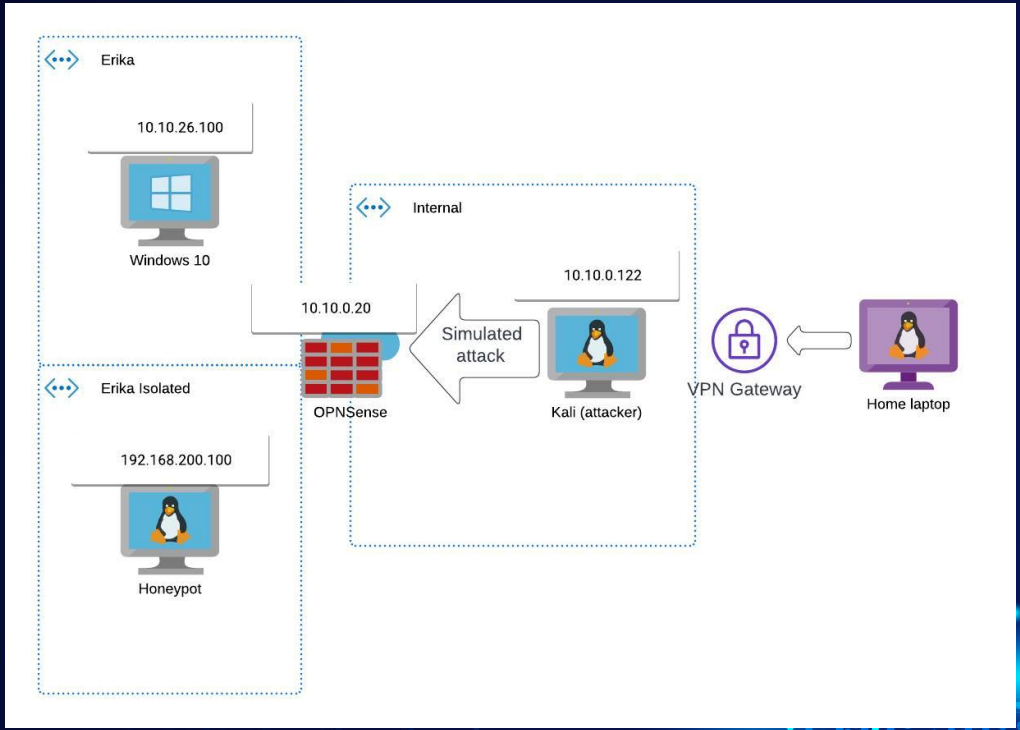
```
  GNU nano 6.2
[Unit]
Description=ENDLESSH demo service
After=network.target
StartLimitIntervalSec=0
[Service]
Type=simple
Restart=always
RestartSec=1
User=endlessh
ExecStart=/usr/local/bin/endlessh -v

[Install]
WantedBy=multi-user.target
```

- Creating new users -> LEAST PRIVILEGE!
- Creating new files: sudo touch <name>.service

04 Securing it

Architecture



OPNsense

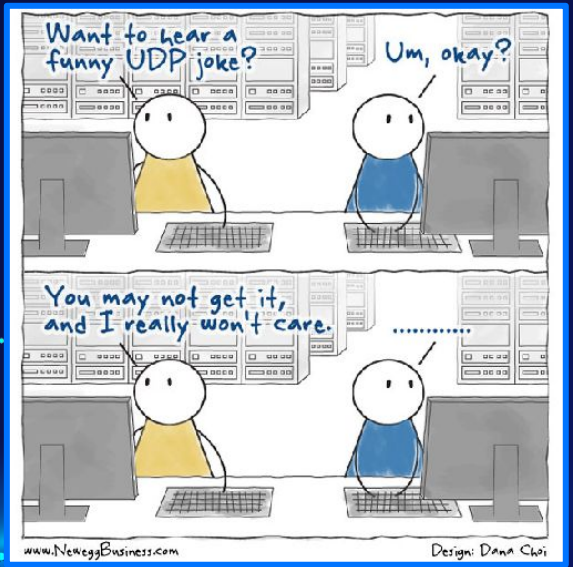
- OPNsense firewall in front of the networks
- A windows workstation on the “corporate LAN”
- Honeypot is on isolated LAN
- 22, 2222, 2223, 2224 ports forwarded to honeypot

  (regular SSH, endlessh, ssh-auth-logger, cowrie)
- Outbound only 516/udp is open for log forwarding

05 Log collection

Overview

- Used syslog-ng as it was already set up
- System logs included from honeypot VM
- Pushes logs on 516/udp to syslog-ng store box
- Store box forward logs to splunk

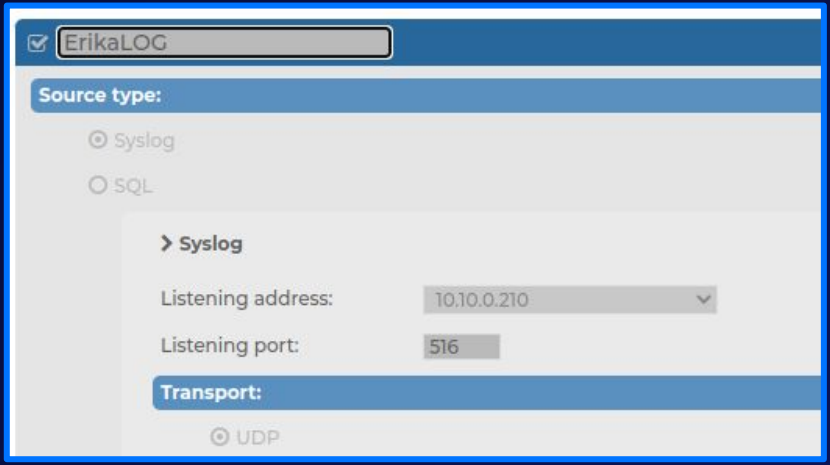


Syslog-ng and Splunk

- ➔ Install syslog-ng-core on SLD-U-Erika
- ➔ Check Log / Logspaces / ErikaLOG (See the screenshot)

  10.10.0.210
- ➔ Configure the syslog-ng.conf (Screenshot on the next slide!)
- ➔ Grab and go the relevant data on Splunk

  10.10.0.166:8000 -> “index = erikaindex” and “endlessh” / “ssh-auth-logger”/ “cowrie”



```
@version:3.35
@include "scl.conf"

options {flush_lines (0); keep_hostname (yes);};

source s_sys { system(); internal();};
source s_cowrie { file("/home/cowrie/cowrie/var/log/cowrie/cowrie.log" program-o
verride("cowrie")); };
destination d_mesg { file("/var/log/messages"); };
destination d_logserver { network("10.10.0.210" transport(udp) port(516)); };
filter f_default { level(info..emerg) and not (facility(mail)); };

log {
        source(s_sys);
        source(s_cowrie);
        destination(d_mesg);
        destination(d_logserver);
 };
```

06 Demo time!

Port scanning

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ nmap -p2222-2224 10.10.0.20 -Pn
Starting Nmap 7.93 ( https://nmap.org ) at 2023-11-26 14:34 CET
Nmap scan report for 10.10.0.20
Host is up (0.00094s latency).

PORT     STATE SERVICE
2222/tcp open  EtherNetIP-1
2223/tcp open  rockwell-csp2
2224/tcp open  efi-mg

Nmap done: 1 IP address (1 host up) scanned in 0.15 seconds

┌──(erika㉿SLD-Kali-Erika)-[~]
└─$
```

Interact with Endlessh

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ ssh root@10.10.0.20 -p 2222 -v

OpenSSH_9.3p2 Debian-1, OpenSSL 3.0.10 1 Aug 2023
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to 10.10.0.20 [10.10.0.20] port 2222.
debug1: Connection established.
debug1: identity file /home/erika/.ssh/id_rsa type -1
debug1: identity file /home/erika/.ssh/id_rsa-cert type -1
debug1: identity file /home/erika/.ssh/id_ecdsa type -1
debug1: identity file /home/erika/.ssh/id_ecdsa-cert type -1
debug1: identity file /home/erika/.ssh/id_ecdsa_sk type -1
debug1: identity file /home/erika/.ssh/id_ecdsa_sk-cert type -1
debug1: identity file /home/erika/.ssh/id_ed25519 type -1
debug1: identity file /home/erika/.ssh/id_ed25519-cert type -1
debug1: identity file /home/erika/.ssh/id_ed25519_sk type -1
debug1: identity file /home/erika/.ssh/id_ed25519_sk-cert type -1
debug1: identity file /home/erika/.ssh/id_xmss type -1
debug1: identity file /home/erika/.ssh/id_xmss-cert type -1
debug1: identity file /home/erika/.ssh/id_dsa type -1
debug1: identity file /home/erika/.ssh/id_dsa-cert type -1
debug1: Local version string SSH-2.0-OpenSSH_9.3p2 Debian-1
debug1: kex_exchange_identification: banner line 0: D}}'-
```

Passwords used for brute force

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ cat passwords.txt
test
dev
devdev
devdevdev
qa
god
admin
adminadmin
admins
goat
sysadmin
water
dirt
air
earth
company
company1
company123
company1!
company!
secret
```

Brute force SSH-auth-logger with hydra

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ hydra -s 2223 -l root -P passwords.txt 10.10.0.20 ssh -V -t 4 -w 3
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military o
ses (this is non-binding, these *** ignore laws and ethics anyway).

[WARNING] the waittime you set is low, this can result in errornous results
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-26 14:39:47
[DATA] max 4 tasks per 1 server, overall 4 tasks, 101 login tries (l:1/p:101), ~26 tries
[DATA] attacking ssh://10.10.0.20:2223/
[ATTEMPT] target 10.10.0.20 - login "root" - pass "test" - 1 of 101 [child 0] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "dev" - 2 of 101 [child 1] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "devdev" - 3 of 101 [child 2] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "devdevdev" - 4 of 101 [child 3] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "qa" - 5 of 101 [child 0] (0/0)
```

Brute force on Cowrie

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ hydra -s 2224 -l root -P passwords.txt 10.10.0.20 ssh -V -t 4 -w 3
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military o
ses (this is non-binding, these *** ignore laws and ethics anyway).

[WARNING] the waittime you set is low, this can result in errornous results
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-26 14:43:40
[WAR
g,
[DAT
[DAT
[ATT
[ATT
[ATT
```

```
[ATTEMPT] target 10.10.0.20 - login "root" - pass "monkey" - 94 of 101 [child 1] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "letmein" - 95 of 101 [child 2] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "login" - 96 of 101 [child 3] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "princess" - 97 of 101 [child 0] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "solo" - 98 of 101 [child 1] (0/0)
[ATTEMPT] target 10.10.0.20 - login "root" - pass "qwertyuiop" - 99 of 101 [child 2] (0/
[ATTEMPT] target 10.10.0.20 - login "root" - pass "starwars" - 100 of 101 [child 3] (0/0
[ATTEMPT] target 10.10.0.20 - login "root" - pass "password" - 101 of 101 [child 0] (0/0
[2224][ssh] host: 10.10.0.20   login: root   password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-26 14:44:59

┌──(erika㉿SLD-Kali-Erika)-[~]
└─$
```

Cowrie fake shell

```
┌──(erika㉿SLD-Kali-Erika)-[~]
└─$ ssh root@10.10.0.20 -p 2224 -v

OpenSSH_9.3p2 Debian-1, OpenSSL 3.0.10 1 Aug 2023
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no fil
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to 10.10.0.20 [10.10.0.20] port 2224.
debug1: Connection established.
debug1: identity file /home/erika/.ssh/id_rsa type -1
debug1: identity file /home/erika/.ssh/id_rsa-cert type -1
debug1: identity file /home/erika/.ssh/id_ecdsa type -1
debug1: identity file /home/erika/.ssh/id_ecdsa-cert type -1
debug1: identity file /home/erika/.ssh/id_ecdsa_sk type -1
```

Check network

```
root@svr04:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 1c:68:8a:7a:56:e4
          inet addr:192.168.200.100  Bcast:192.168.200.255  Mask:255.255.255.0
          inet6 addr: fe77::175:e5ff:fe48:d401/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:398495 errors:0 dropped:0 overruns:0 frame:0
          TX packets:522134 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:506025225 (506.0 MB)  TX bytes:32385103 (32.4 MB)


lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:110 errors:0 dropped:0 overruns:0 frame:0
          TX packets:110 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:18589297 (18.6 MB)  TX bytes:18589297 (18.6 MB)
```

Log collection with Splunk

i	Time	Event
>	11/16/23 9:17:57.000 PM	<30>Nov 16 21:17:57 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"eot","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6.1"," host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw
>	11/16/23 9:17:55.000 PM	<30>Nov 16 21:17:55 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"citi","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6.1", host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw
>	11/16/23 8:09:22.000 PM	<30>Nov 16 20:09:22 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"gertrer","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6. host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw
>	11/16/23 8:09:21.000 PM	<30>Nov 16 20:09:21 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"valami","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6.1 host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw
>	11/16/23 7:44:26.000 PM	<30>Nov 16 19:44:26 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"again","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6.1" host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw
>	11/16/23 7:44:24.000 PM	<30>Nov 16 19:44:24 SLD-U-Erika ssh-auth-logger[14532]: {"client_version":"SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.3","destinati l":"info","msg":"Request with password","password":"whatever","product":"ssh-auth-logger","server_version":"SSH-2.0-libssh-0.6 host = 10.10.0.210 source = 10.10.0.166:601 sourcetype = tcp-raw

Next steps / opportunities

- Ansible for automation
- Dockerize the services
- Allow outgoing traffic for more insights
- New kind of honeypots

Links

1. https://github.com/skeeto/endlessh
2. https://github.com/cowrie/cowrie
3. https://github.com/JustinAzoff/ssh-auth-logger
4. https://github.com/paralax/awesome-honeypots
5. https://www.geeksforgeeks.org/how-to-use-hydra-to-brute-force-ssh-connections/
6. https://nmap.org/book/man-port-specification.html
7. https://journey.study/v2/learn/courses/219/modules/472/units/0
8. https://serverastra.com/docs/Tutorials/Setting-Up-and-Securing-SSH-on-Ubuntu-22.04%3A-A-Comprehensive-Guide
9. https://linuxconfig.org/how-to-enable-and-disable-ssh-for-user-on-linux
10. https://www.ibm.com/docs/en/db2/11.1?topic=installation-enable-disable-remote-root-login
11. https://cowrie.readthedocs.io/en/latest/INSTALL.html
12. https://github.com/cowrie/cowrie/issues/1614
13. https://cowrie.readthedocs.io/en/latest/systemd/README.html
14. https://axoflow.com/syslog-over-udp-message-loss-1/

Thank you for your attention!