Raspberry PI

malware honeypot

AGENDA

* honeypot, what is it? what is it's purpose?

* Dionaea honeypot, hardware, software and network setup

* MHN server, how to manage sensors effectively?

* system security, quick demo, todo and links

TL;DR

* Raspberry PI runs Dionaea honeypot for 2 months

* it is connected to our guest WiFi network

* and it catched nothing ;-]

HONEYPOT

WHAT IS HONEYPOT?

* honeypot is a decoy infrastructure that is deployed to be attacked

* since it has no other purpose, every attempt to interact is suspicious

WHAT IS IT'S PURPOSE?

* it can discover malicious activity, especially when deployed behind firewall

* it can slow down and mislead the attacker by providing slow responses or incorrect information

* it can collect the logs, tools and other stuff left by attacker to aid forensics

MY GOALS

* excuse to purchase Raspberry PI ;-]

* monitor malware activity in our network, as ½ of the office runs on Windows

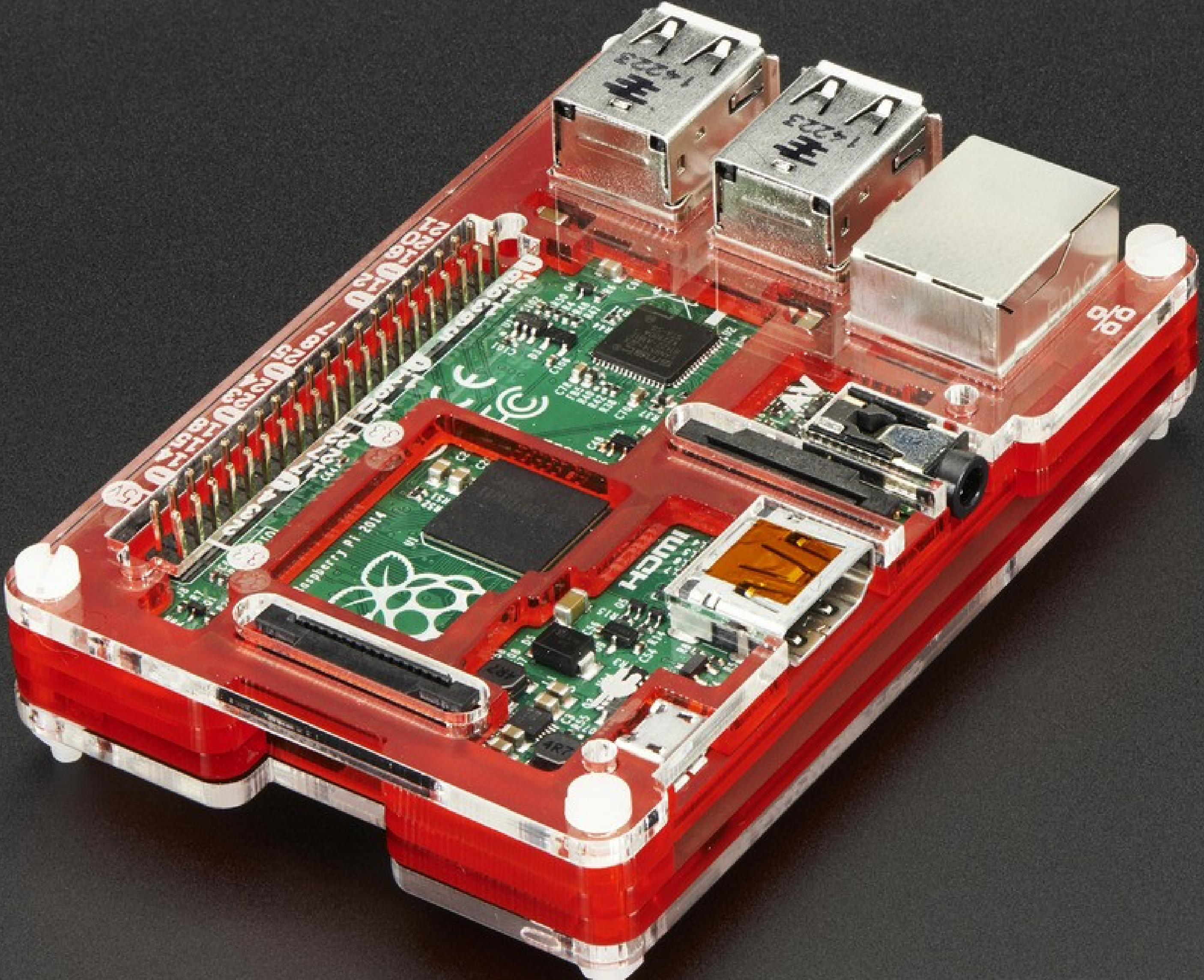
* detect network scans, as our guest WiFi isn't separated

DIONAEA HONEYPOT

HARDWARE

* cheap, affordable, yet fashionable Raspberry PI 2
    * 900MHz quad-core ARM Cortex-A7 CPU
    * 1GB RAM
    * 8GB SD card
    * WiFi dongle

* costs around €65 (inc. Sue's fee;)



SOFTWARE

* Raspbian, optimized Debian for the Raspberry PI

* Dionaea, low interaction, malware detection and network reconnaissance honeypot

HOW DIONAEA WORKS?

* it listens on few ports and reports connections

* it waits for malware and pretends to be exploitable

* it will capture and store payload

NETWORK SETUP

* wlan0, auto DHCP, connects on boot to Guest WiFi

* headless setup, plug in 5V and SSH into it 30s later

* current IP and MAC address are [REDACTED ;-]

MORE DETAILS

* opened ports: 21 (FTP), 22 (SSH), 42 (WINS), 69 (TFTP), 135 (RPC), 445 (SMB), 1433 (MSSQL), 3306 (MYSQL), 5060/5061 (SIP/VOIP)

* hostname: accounting-dev

* detectability: spoof MAC address to imitate DELL, move SSH to non-standard port, attempt would trigger alert anyway

MANAGEMENT SERVER

MHN (Modern Honey Network)

* so if honeypots are so cool why we don't use them?

* they are difficult to deploy and maintain

* also often log to files

MHN (Modern Honey Network)

* guys from ThreatStream developed MHN, open-source honeypot management server

* automates deployment process

* sets up data flows with hpfeeds

* collects data and correlates it with GeoIP

* does real time visualization with honeymap

SYSTEM SECURITY

* MHN is running on t2.micro EC2 instance

* access to server is restricted to office and home IPs

* no extra services, only necessary ports are opened

* SSH keys everywhere, self-signed SSL cert ;-]

* regular updates are necessary

DEMO

TODO

TODO

* migrate MHN to proper server

* integrate with Slack or centralized logging

* persuade CTO to buy 3 boards ;-]

TODO continued

* use OpenVPN for consistent IPs

* deploy more sensors, ex. Kippo, high interaction SSH honeypot

* script updates, (MHN rules, Diaonea package)

GITHUB

GITHUB

* github.com/rep/dionaea

* github.com/desaster/kippo

* github.com/threatstream/mhn

THE END