A JOURNEY INTO DECEPTION BASED SECURITY

ADEL KARIMI AND ELLIOTT BRINK

BSIDES CANBERRA
18TH MARCH 2017

Photo: Adel Karimi

/usr/bin/whoami

- Adel Karimi <@0x4d31>
  - Senior Security Engineer
  - Member of the Honeynet Project since 2010
    - Chapter lead (2010-2017)
  - Started Trapbits Honeypot Community in April 2016
    - Join us @trapbits <trapbits.github.io>
  - Hobbyist {astro}photographer

/usr/bin/whoami

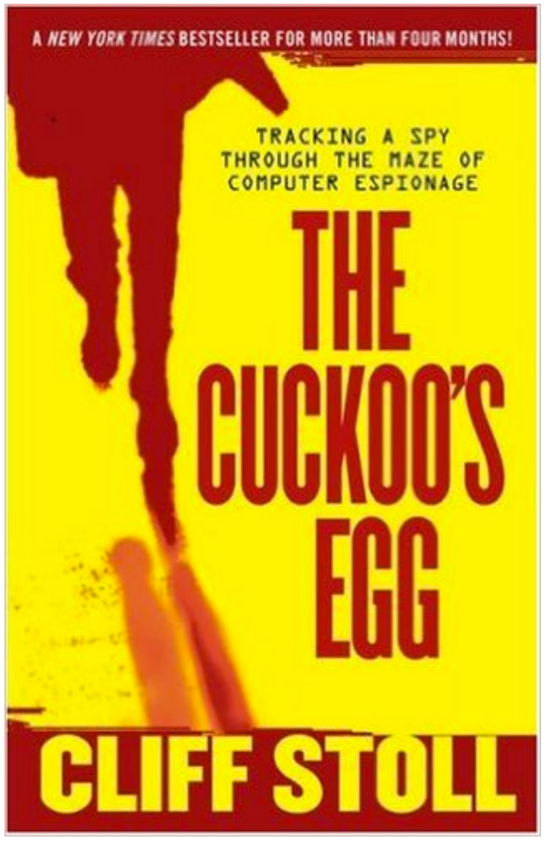
- Elliott Brink <@ebrinkster>
  - Senior IT Security Consultant @ RSM
  - I do:
    - Internal Penetration Testing
    - External Penetration Testing
    - Social Engineering
  - Speaker at various information security/hacker conferences:
    - DEFCON 23 WoS, GrrCon, BSides Canberra+Indianapolis, others.
  - Former top 10 consulting, sysadmin
  - Honeypot crazy (coworkers/friends agree)

OUTLINE

- Part 1
  - An intro to deception
  - Honeypot
    - Concept and use-cases
    - Different types
    - Honeytokens
  - Honeypots in production environment
  - Deception as a breach detection tool
- Part 2
  - Hands-on Honeypotting

INTRO

- Clifford Stoll's interesting story of stalking the wily hacker back in the 80s!
- Technology has changed a lot, but the concept of honeypots and deception in general has remained the same
- Mostly used by research community for discovering new attacks, collecting malware, and studying the attacker's tools, tactics and motivations
- … still not widely accepted and deployed in production environments

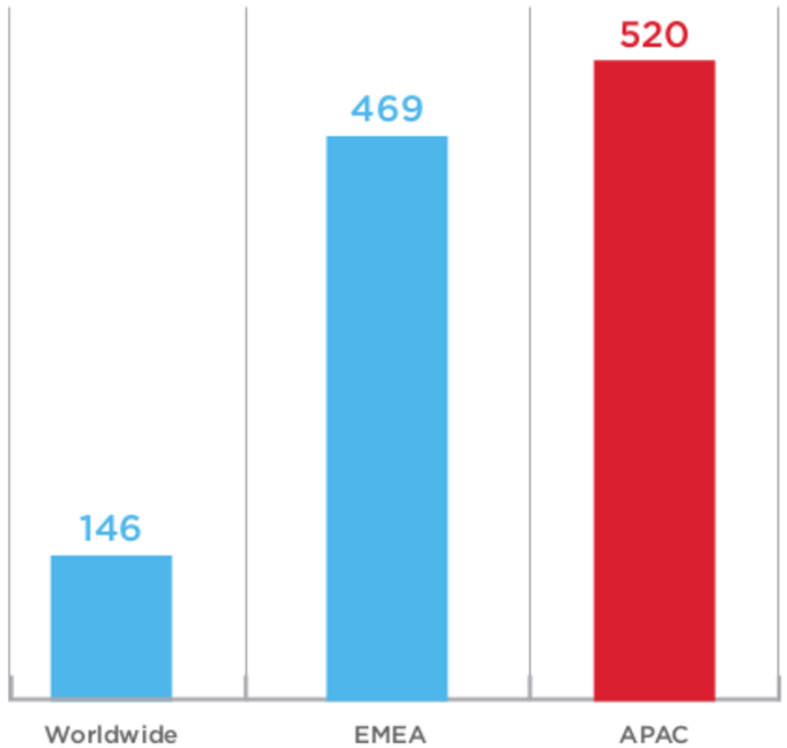


The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage

INTRO II

- Enterprises have shifted their security focus from prevention to detection and response ~> **Breach detection** became a new must-have security tool

- Median time from compromise to discovery



M-Trends 2016: https://www.fireeye.com/current-threats/annual-threat-report/mtrends.html

INTRO III

- Deception for *{post} breach detection*
- It's FREE, although… *{CYBER deception vendor$..}*
- Gartner: by 2018 about %10 of enterprises will use deception tools and techniques against attackers

JUST FOR FUN!

- Definition of “honeypot” and “deception” by one of the CYBER deception vendors:
  - Unlike a honeypot (an early stage form of deception), which was designed to be a low interaction honeypot for detecting automated scanning tools and worms, deception is designed to detect inside-the-network threats and the lateral movement by human attackers.

"All warfare is based on **deception**."

–SUN TZU <THE ART OF WAR>

Sergio Caltagirone
@cnoanalysis

Following

TTST (Time to Sun Tzu) - the amount of time passed from the start of an #Infosec conference to the first Sun Tzu quote in a presentation

RETWEETS 31 LIKES 63

8:33 PM - 12 Sep 2016

MILITARY DECEPTION

- Deception as a military strategy
  - refers to attempts mislead enemy forces during warfare
- Operation bodyguard: World War II deception plan
  - Operation Fortitude, Operation Quicksilver, etc.
  - Operation Skye: radio deception component of Fortitude North, involving simulated radio traffic between fictional army units
  - Fictional field armies
  - Faked operations
  - False "leaked" information
  - Inflatable tanks
  - Soundtracks
  - Fake radio transmissions

MILITARY DECEPTION II

DENIAL AND DECEPTION (D&D)

- A theoretical framework for conceiving and analyzing military intelligence techniques pertaining to secrecy and deception.
- Originating in the 1980s, it is roughly based on the more pragmatic Soviet practices of maskirovka
- Maskirovka (Russian military deception)

Measure	Western Equivalent	Techniques	Example
Concealment	Camouflage	Awnings, Smoke Screens, Nets, Radio Silence	Building Tanks In An Automobile Plant
Imitation	Mimicry	Decoys, Military Dummies	Dummy Tanks With Radar Reflectors; Decoy Bridges Created By A Line Of Floating Radar Reflectors
Simulation	Simulation	Decoys, Etc.	Dummy Artillery Battery Complete With Noise And Smoke
Disinformation	Disinformation		False Letters; Untrue Information To Journalists; Inaccurate Maps; False Orders; Orders With False Dates
Demonstrative Manoeuvres	Feints	False Trails	Attacks Away From The Main Thrust; Pontoon Bridges Away From Attack Routes

https://en.wikipedia.org/wiki/Russian_military_deception
https://en.wikipedia.org/wiki/Denial_and_deception

D&D METHODS MATRIX

Deception Objects:	Deception: Mislead-Type Methods	Denial: Ambiguity-Type Methods
Action:	Revealing	Concealing
Fact	reveal facts: nonessential elements of friendly information (NEFI) • *publish true information to support your deception story* • *reveal deception capabilities ~> makes the attackers disbelieve info collected from post-compromise. it also has a deterrent effect*	conceal facts (dissimulation): essential elements of friendly information (EEFI) • hide software using stealth methods • deny access to system resources
Fiction	reveal fictions (simulation): essential elements of deception information (EEDI) • expose fictional systems • allow disclosure of fictional information	conceal fictions: non disclosable deception information (NDDI) • keep deceptive security operations a secret • hide simulated information on honeypots

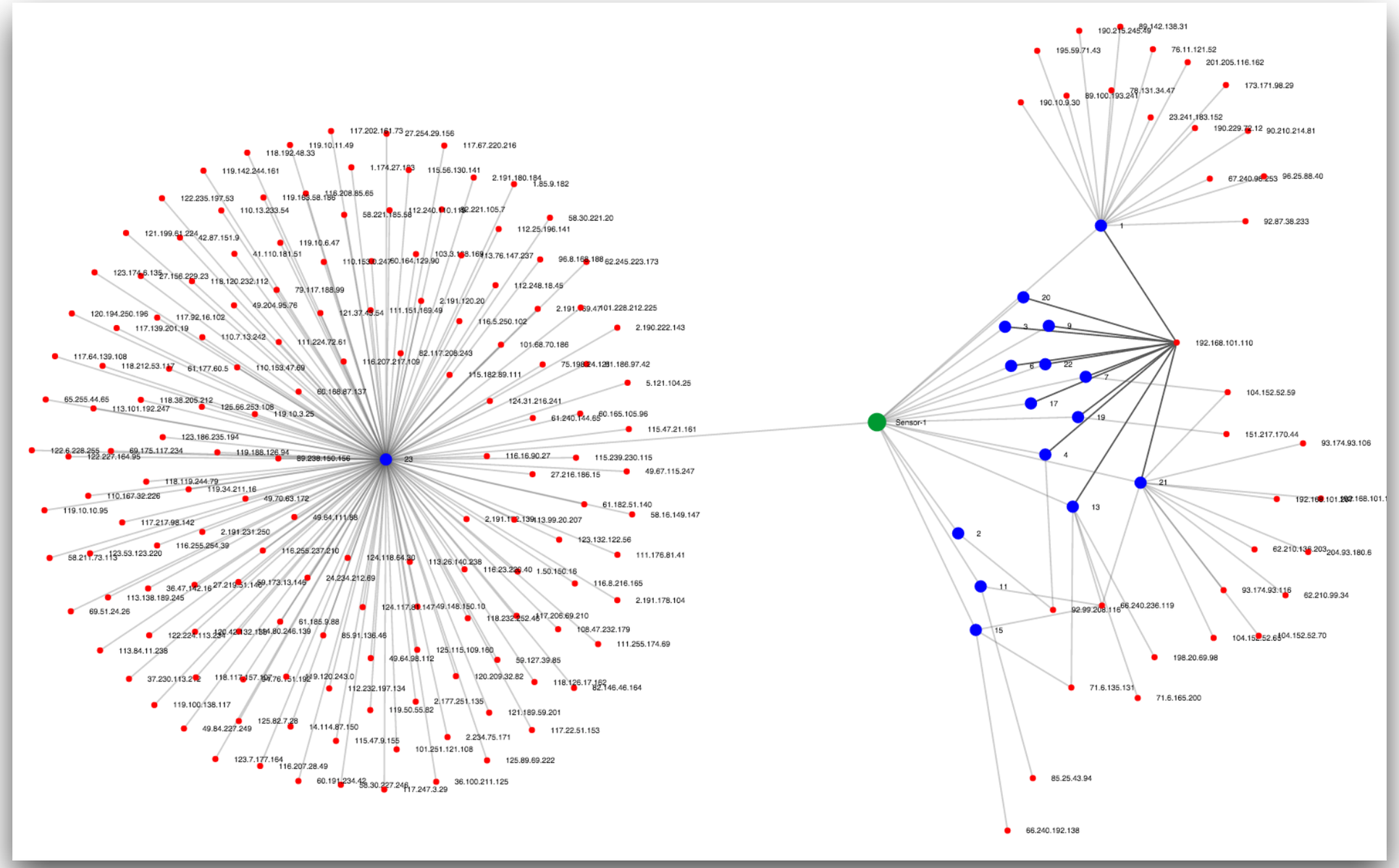
Bennett and E. Waltz. Counterdeception Principles and Applications for National Security. Artech House: Boston, MA, 2007.

DECEPTION COMPONENTS

- Honeypots or Decoys
- Honey tokens/bits
- Monitoring and alerting
- Deception stories

DECEPTION COMPONENTS II

- Data visualization



USE HONEYPOTS TO DECEIVE AND OBSERVE YOUR ENEMIES

HONEYPOT

HONEYPOT

- No signatures, no fancy algorithms
  - Solely based on **deception**
- "Honeypot is a security resource whose value lies in being probed, attacked, or compromised."

  -Lance Spitzner
- Honeypots are not production systems
  - they don't have any authorized use
  - ~> ANY interaction with a honeypot implies malicious or unauthorized activity

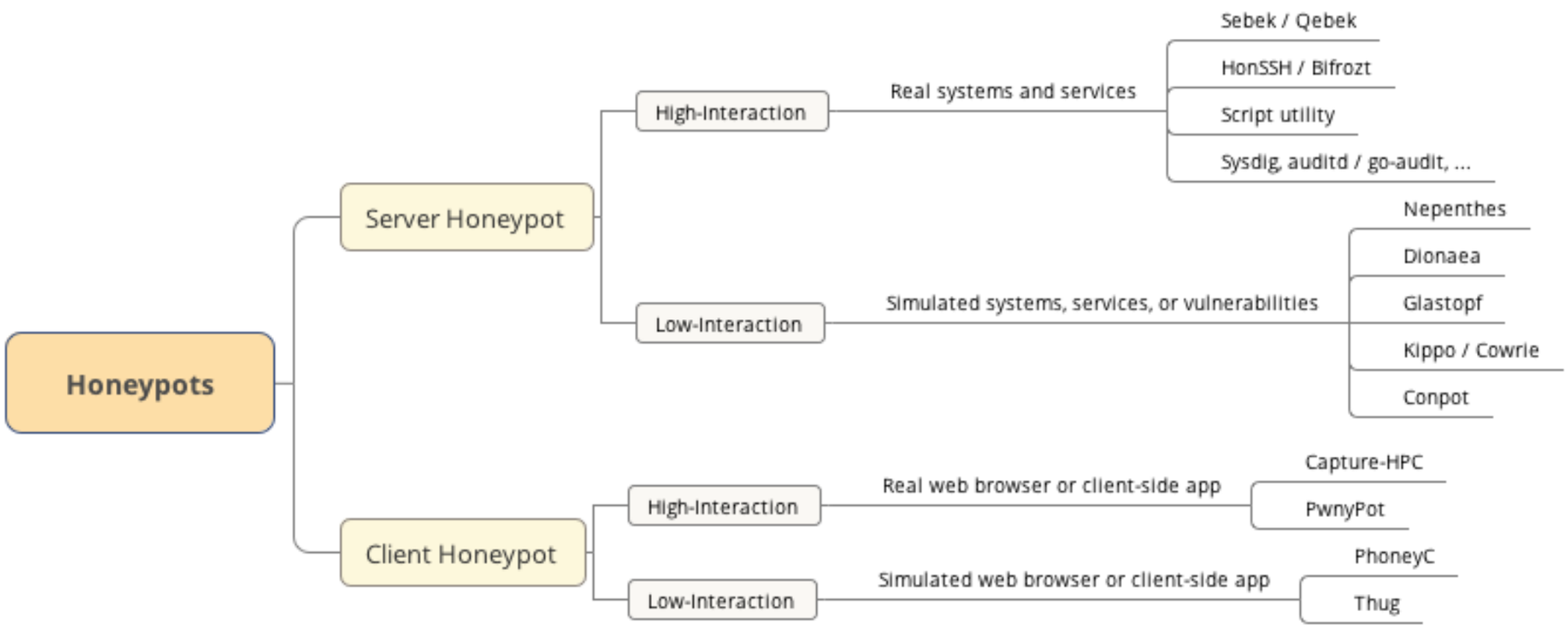
HONEYPOT USE-CASES

- You can use Honeypots to
  - Detect/track botnets
  - Learn new attack methods and trends
  - Collect malware
  - Detect 0-day attacks!?
  - Detect malicious servers (using client honeypots)
  - Delay and misdirect attackers
  - Detect compromised systems
- It depends on how and where you use it

DIFFERENT TYPES

- Low-Interaction vs. High-Interaction
  - Risk, installation and maintenance, data gathering
- Server Honeypot (traditional) vs. Client Honeypot
  - Passive vs. Active
  - Detection mechanism
- Honeytokens
- *Research vs. Production*

DIFFERENT TYPES



DETECTING DRIVE-BY DOWNLOADS AND MALICIOUS SERVERS

THUG DEMO

HONEYTRAP / TRAP / BREADCRUMB / HONEYBITS

HONEYTOKENS

HONEYTOKENS

- Honeytokens can be any resources like a bogus file or a fake database record that – just like the honeypots – don't have any authorized use. So, any interaction with it can be considered as a potential breach or malicious activity.

- Different types:

  1. Monitored bogus resources

  2. Honeytokens that contain beacons that are triggered when they are opened or accessed! ~> not good for catching advanced attackers

     - CanaryTokens

  3. Breadcrumbs / Honeybits: Their value lies in being used (not just accessed) and leading the attackers to your decoys.

     - We don't need to monitor the access to these tokens

LEAD THE ATTACKERS TO YOUR DECOYS/HONEYPOTS

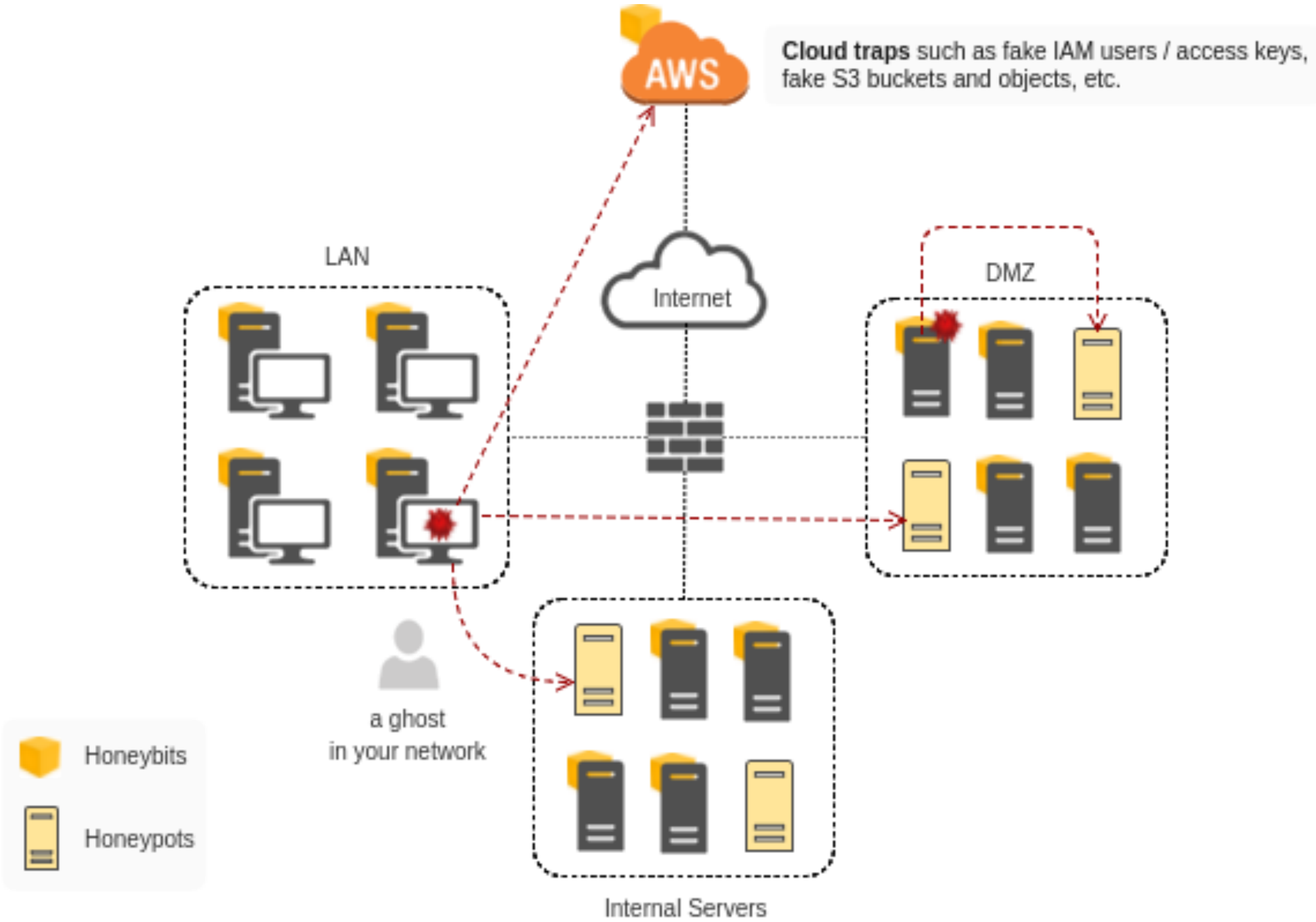
HONEYBITS

HONEYBITS

- A simple tool to create and place breadcrumbs, honeytoken/traps or as I call it "honeybits", to lead the attackers to your decoys/honeypots! https://github.com/0x4D31/honeybits

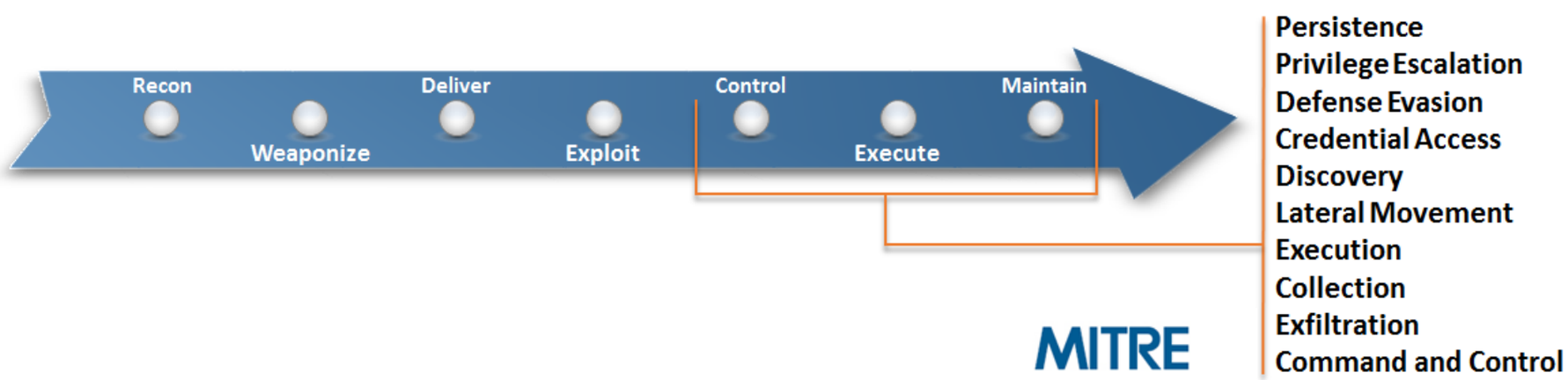
- Features:
  - Insert fake bash_history commands including ssh, ftp, rsync, scp, mysql, wget, awscli
  - Fake AWS credentials and config files
  - Creating honeyfiles and monitoring the access to these traps using auditd or go-audit
  - Content generator for honeyfiles and file honeybits (todo)
    - Configuration, connection and backup files
  - Fake entries in hosts file, ARP table and etc.
  - Remote config using a Remote Key/Value Store such as Consul or etcd

HONEYBITS II



PRODUCTION HONEYPOTS

- The problem with the traditional implementation of honeypots in a production environment
  - Honeypots can only be discovered by network scanning (which is noisy!)
- Cyber Kill-Chain
- ATT&CK Tactic Categories



TL;DR

- The more you plant false or misleading information in response to the post-compromise techniques (especially the techniques under ‘credential access,’ ‘Discovery,’ and ‘Lateral movement’ tactics in ATT&CK matrix), the greater the chance of catching the attackers.

DECEPTION PLANNING

- Specify deception goal
- Identify the attackers' biases / collect threat information
- Design deception story and tactics
- Implement deception components
- Monitor and so on..

DECEPTION STORY

- Describing deception stories using the structure used by TDD/BDD*

```
/*
Deception Story [DS2]: The attacker gains access
    to the database using bogus user accounts
Biases exploitation: Confirmation Biases (Look for
     methods that confirm that there is a weak
    password)
Associated deception tactics: T1
*/
Given an attacker guesses the password of the
      bogus account (brute force attack)
When  the attacker authenticates in the DBMS
Then  an alert is sent to management system
  And the system connects the account with bogus
      databases
```

```
        database using bogus user accounts
Then    an alert is sent to management system
```

* C. De Faveri and A. Moreira, "Designing Adaptive Deception Strategies," 2016 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), Vienna, 2016

AWS S3 TRAP

DEMO

HONEYPOT CHALLENGES

- Honeypot detection/evasion techniques, e.g.:
  - {Escape from monkey island: Evading high-interaction honeyclients} paper
  - {Breaking Honeypots For Fun And Profit} at BH2015
- New attack vectors / platforms
- …

HONEYPOT CHALLENGES

A client honeypot evasion method in Nuclear Pack, April 2012

```
document.onmousemove = function ()
{
  if (window.xyzflag === 0)
  {
    window.xyzflag = 1;
    var head = document.getElementsByTagName("head")[0];
    var script = document.createElement("script");
    script.type = "text/javascript";
    script.onreadystatechange = function ()
    {
      if (this.readyState == "complete")
      {
        window.xyzflag = 2;
      }
    };
    script.onload = function ()
    {
      window.xyzflag = 2;
    };
    script.src = url + Math.random().toString().substring(3) + ".js";
    head.appendChild(script);
```

THE FIRST PUBLIC ANNOUNCEMENT

HONEYNET PROJECT WORKSHOP 2017 AT UNSW CANBERRA 15-17 NOVEMBER 2017

canberra2017.honeynet.org

https://500px.com/heathdomanphotography

QUESTIONS?

ADEL KARIMI
THE HONEYNET PROJECT
Twitter Handle: 0x4d31

PART 2 | HANDS-ON

Source of headline image: http://mylesillustration.deviantart.com/art/Winnie-the-Pooh-Fighter-574407124

Honeypot Installation

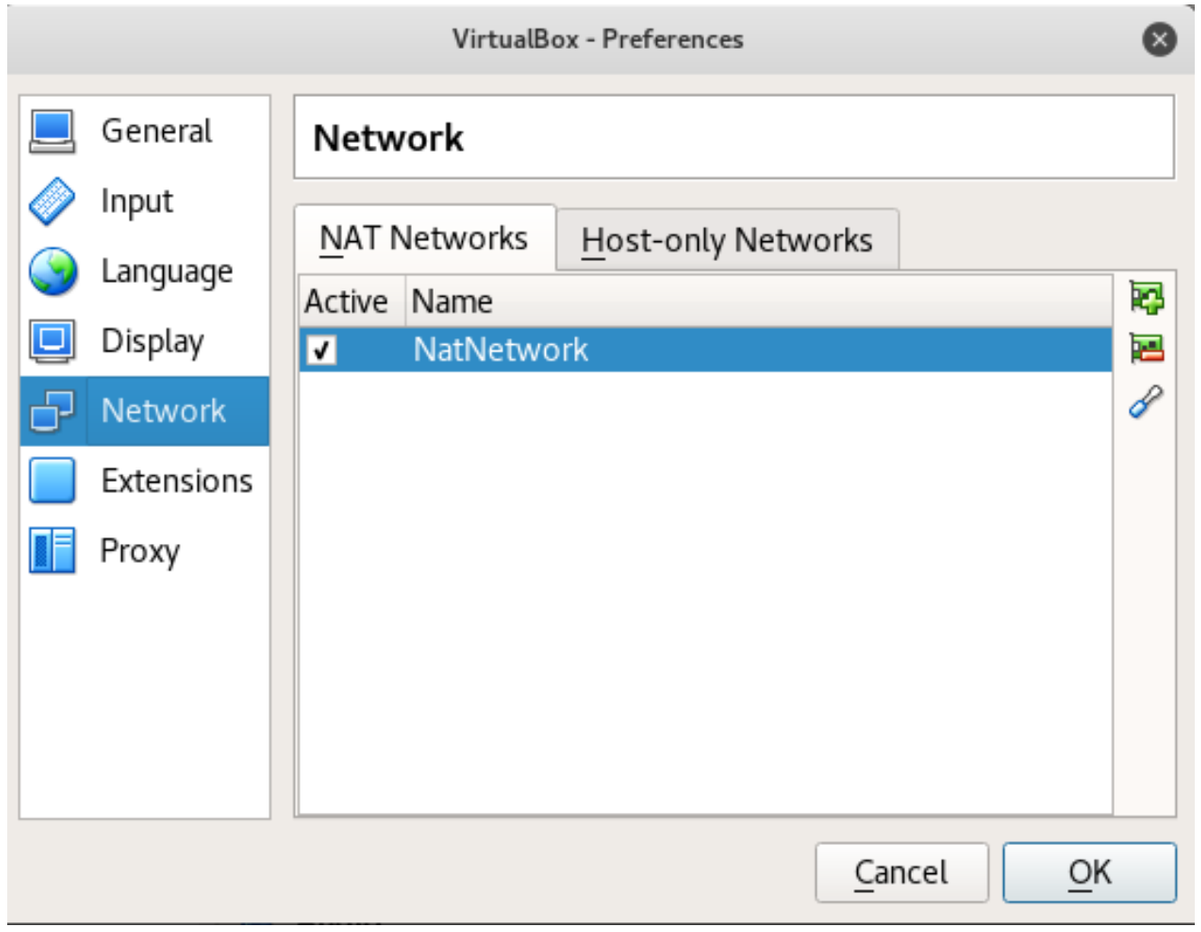
- Blog post with instructions – http://elliottbrink.com/2017/02/20/bsides-canberra-workshop-preparation/
- Download Ubuntu Server 16.04.2 LTS ISO (or workstation)
- Provision the virtual machine with the following specifications:
  - 1 CPU Core
  - 1GB RAM (1024MB)
  - 20GB hard disk space

Ubuntu Workstation Setup

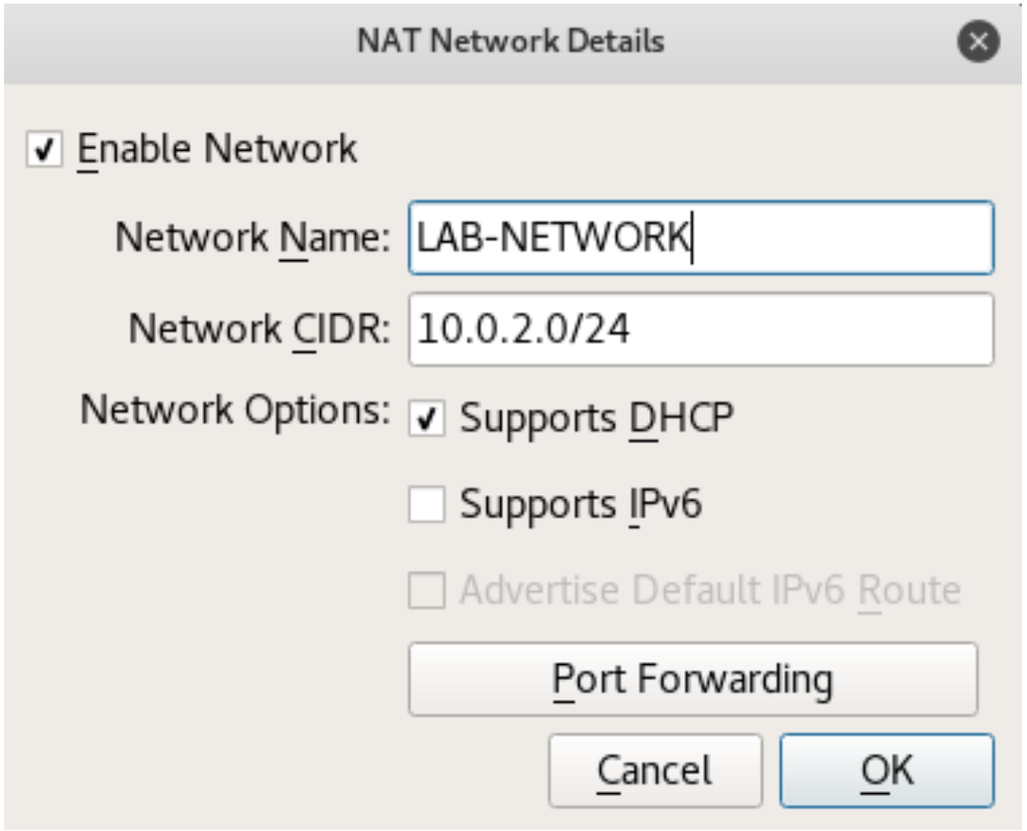
- Let’s install openssh-server and openssh-client

```
sudo apt-get install -y openssh-server
openssh-client
```

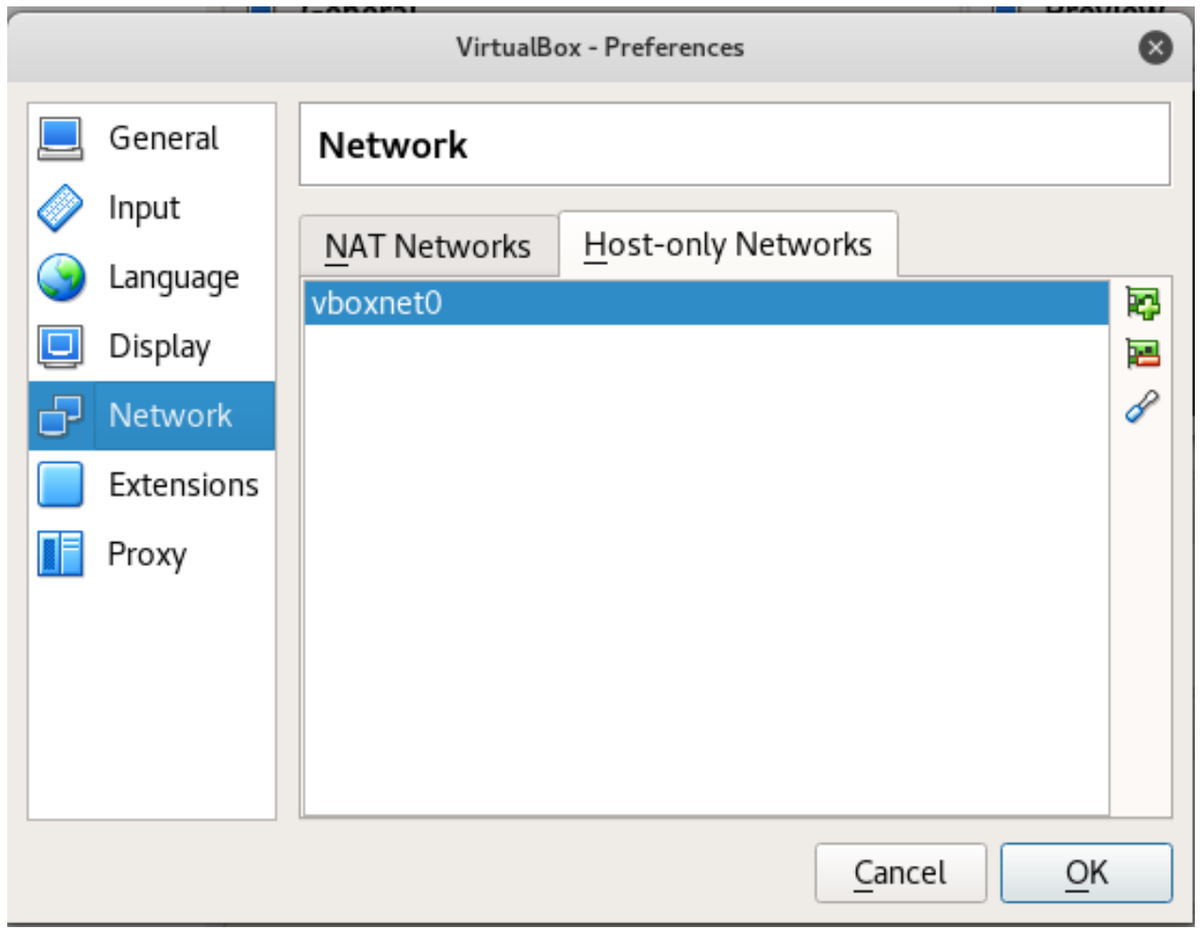
Setup Virtual Networking (System)



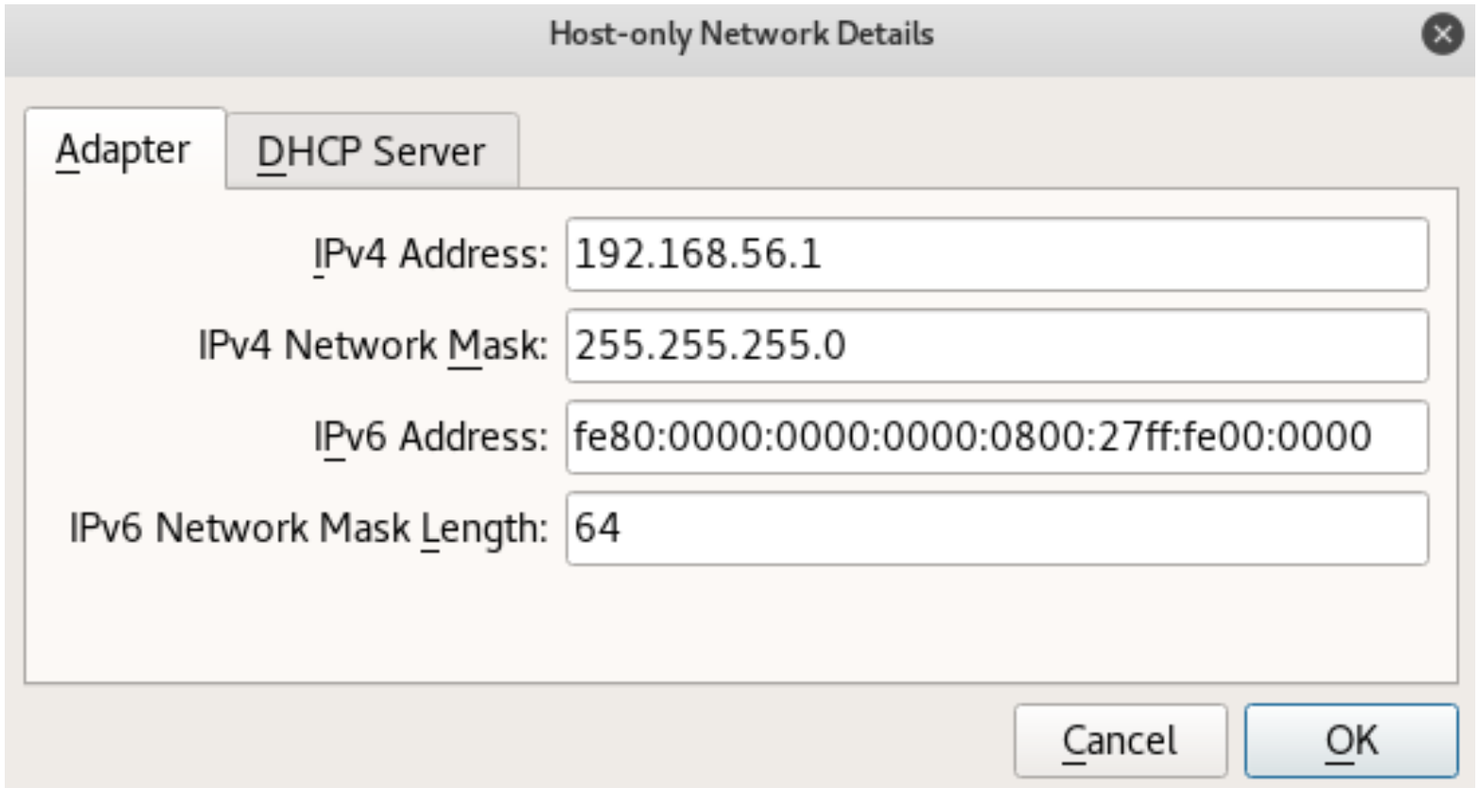
Setup Virtual Networking (System)



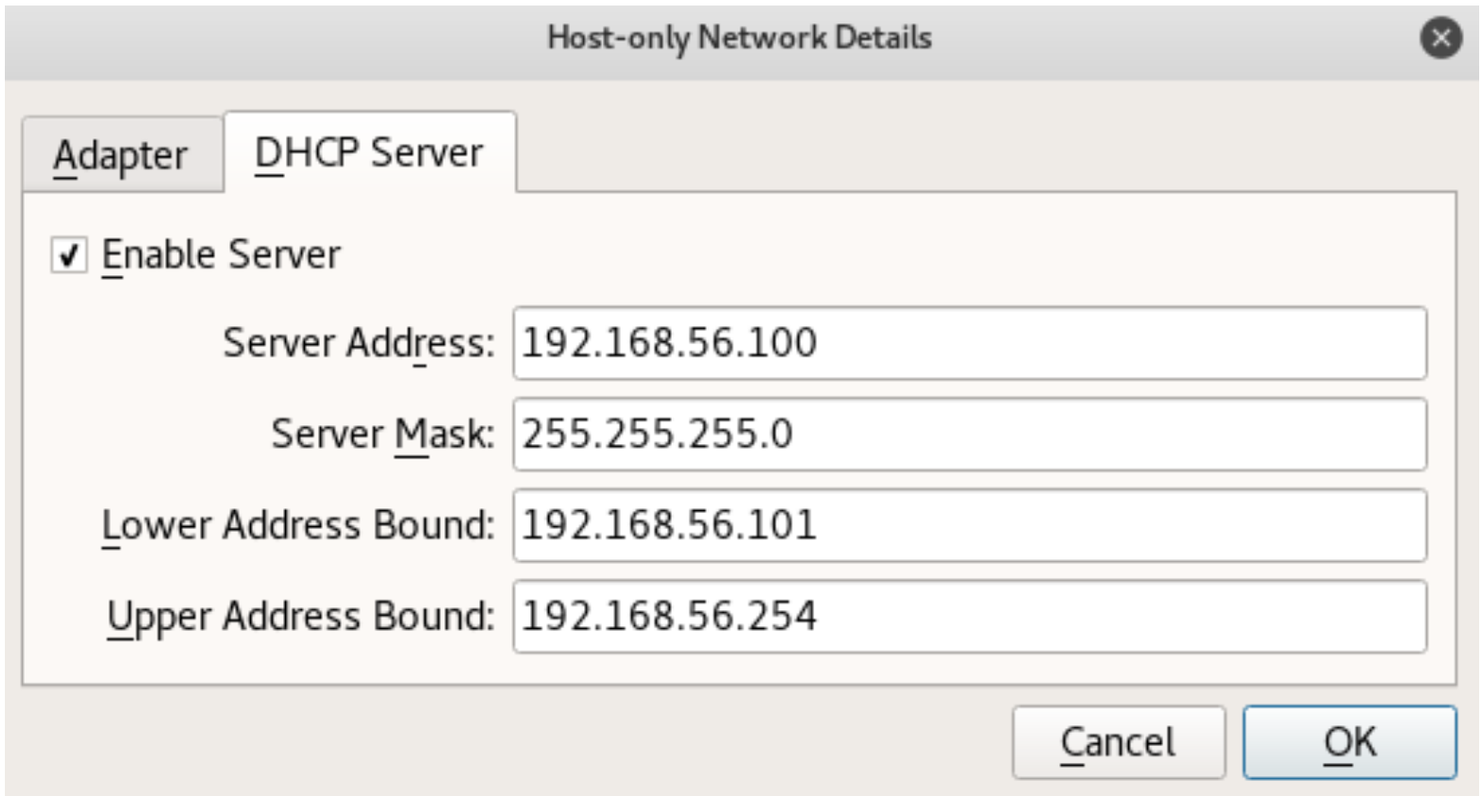
Setup Virtual Networking (System)



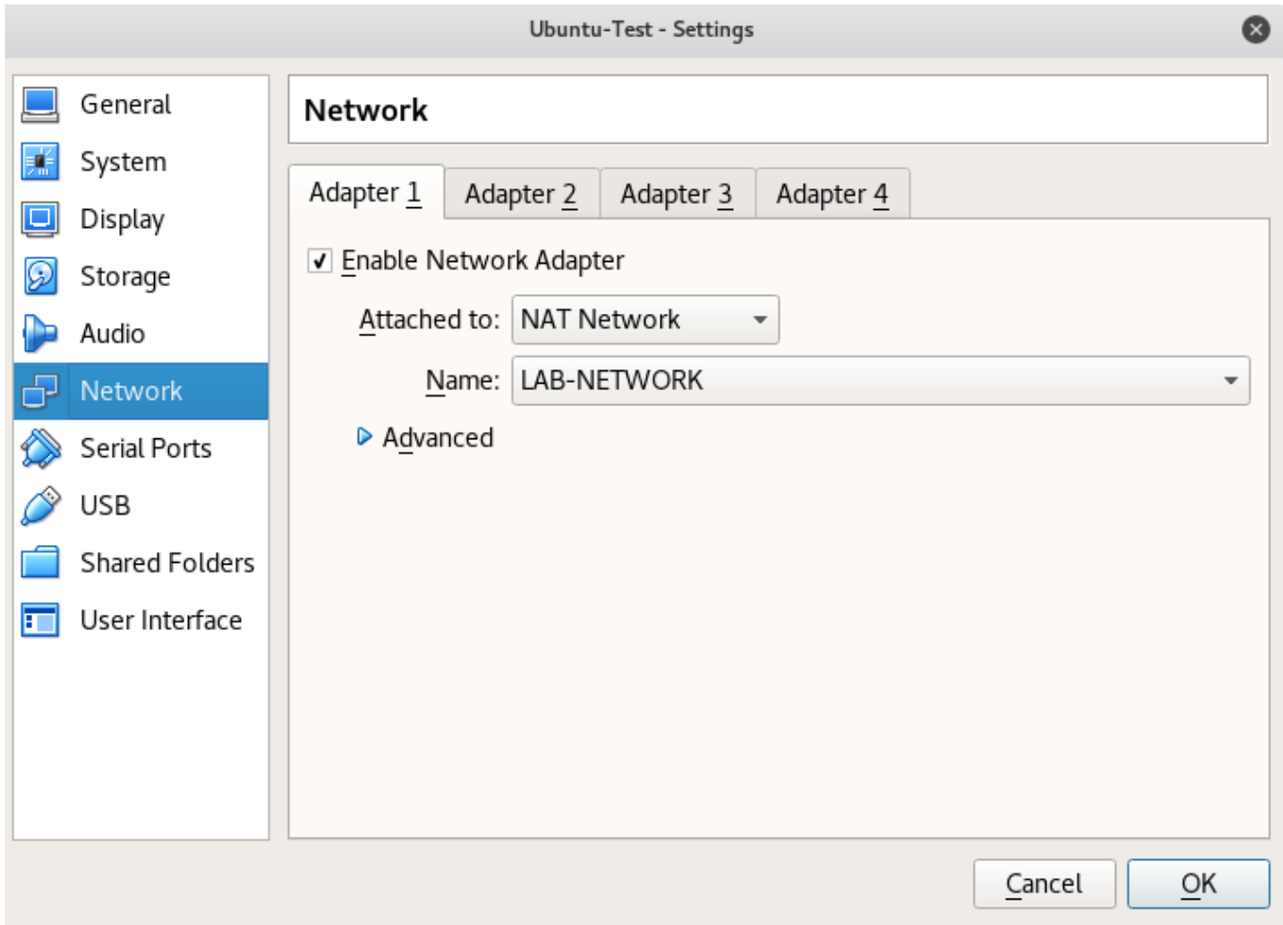
Setup Virtual Networking (System)



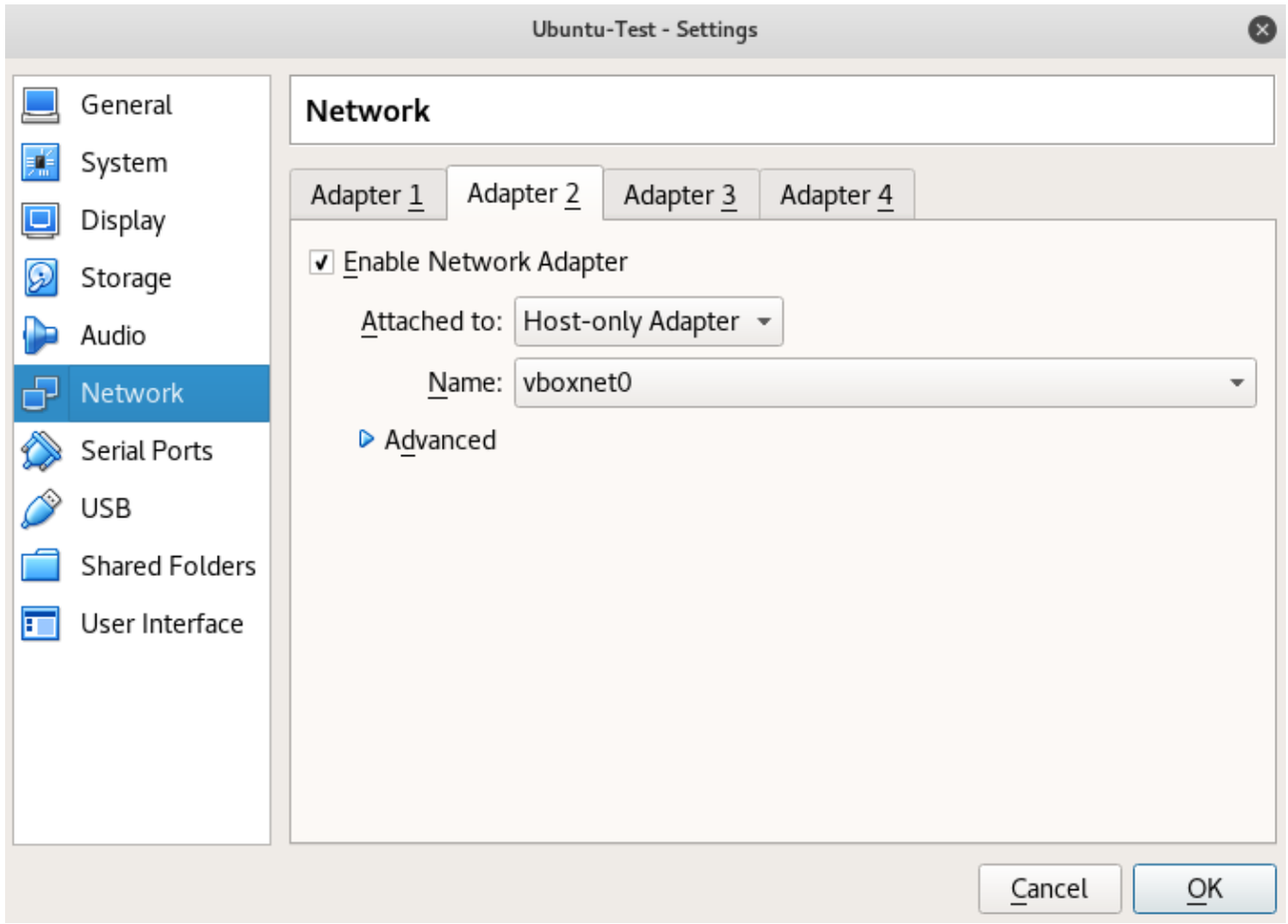
Setup Virtual Networking (System)



Setup Virtual Networking (VM)



Setup Virtual Networking (VM)



Confirm Connection

```
main@main-VirtualBox:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr [redacted]
          inet addr:10.0.2.5  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::f71c:c742:5bae:207e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:78739 errors:0 dropped:0 overruns:0 frame:0
          TX packets:40225 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:112820375 (112.8 MB)  TX bytes:2520608 (2.5 MB)

enp0s8    Link encap:Ethernet  HWaddr [redacted]
          inet addr:192.168.56.102  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::f32a:1128:8766:f8c3/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5592 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4701 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:440468 (440.4 KB)  TX bytes:890642 (890.6 KB)

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:337 errors:0 dropped:0 overruns:0 frame:0
          TX packets:337 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:27656 (27.6 KB)  TX bytes:27656 (27.6 KB)
```

Confirm Connection

```
vboxnet0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.1  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::800:27ff:fe00:0  prefixlen 64  scopeid 0x20<link>
        ether [redacted]  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 169  bytes 29392 (28.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Confirm Connection

```
root@[redacted]:~# ssh main@192.168.56.102
main@192.168.56.102's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

Last login: Mon Feb 20 10:49:05 2017 from 192.168.56.1
main@main-VirtualBox:~$
```

dhclient all the things

- If you’re having any issues, run on your vm:

```
sudo dhclient -v enp0s3;sudo dhclient
-v enp0s8
```

Run my setup script (it's safe I promise, inspect if you like)

```
cd ~;wget https://gist.githubusercontent.com/
ebrinkster/58424ef796eeb8bba844a76bbcc70ab7/
raw/cabff9c7ac7d5f1b0164e242b6d259bc16160098/
honeypot-prep.sh -O honeypot-prep.sh;chmod +x
honeypot-prep.sh;./honeypot-prep.sh
```

https://gist.github.com/ebrinkster

honeypot-prep.sh

Or — http://bit.ly/2mOp5yU

Cowrie Setup

- Script sets it up for you.
- Edit cowrie.cfg and edit the following lines:
  - `hostname = svr04`
  - `[output_textlog]`
  - `logfile = log/audit.log`
  - `format = text`

Cowrie Setup

- Start with ./start.sh
- Test locally on your system
- ssh root@localhost -p2222
- Password: 1234
- You’re running commands on the honeypot
- If you run this in production (be careful) you’ll want to do port forwarding or if it has a public IP do iptables routing
- More info on how to do this on my blog – elliottbrink.com

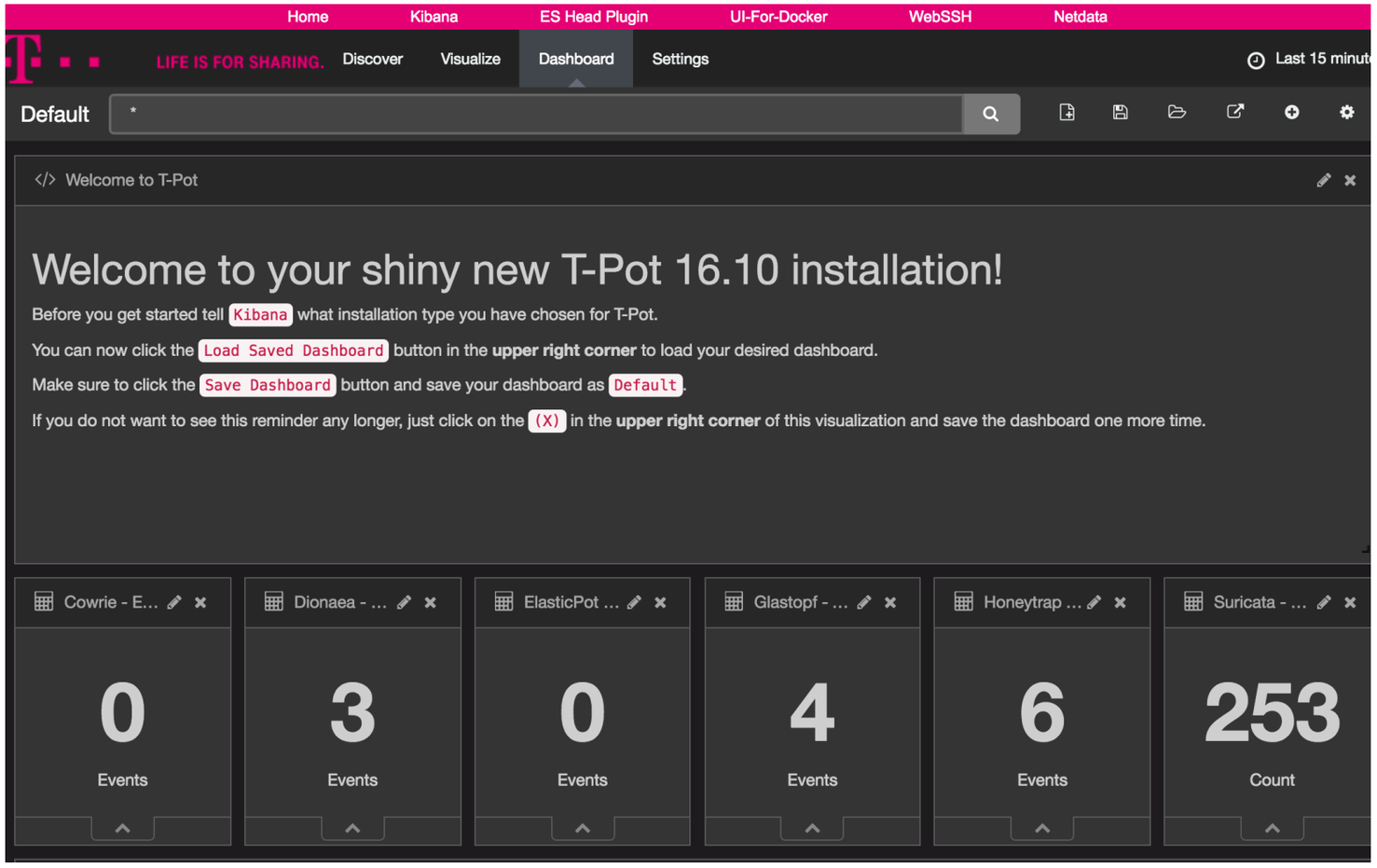
Telnetlogger Setup

- The script sets it up for you, simple make command
- Runs on port 23, need root for it
- sudo ./telnetlogger
- You can edit the C code and edit the port, recompile and do iptables forwarding too.
- Non-interactive, only logs usernames and passwords. Hydra and ncrack, other brute force programs work great against it.

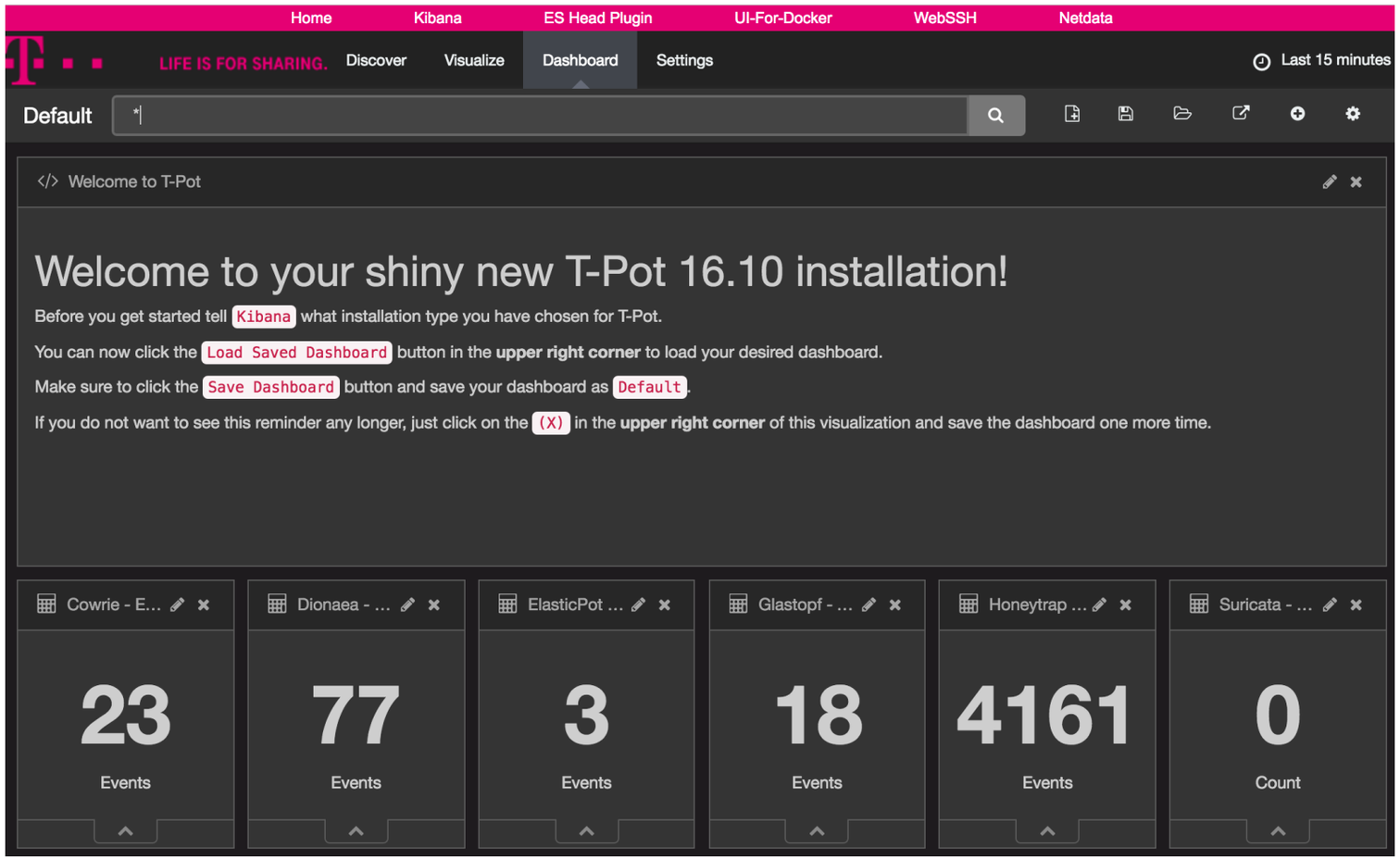
Telnetlogger Setup

```
/*****************************************************************************
 *****************************************************************************/
int
main(int argc, char *argv[])
{
        FILE *fp_passwords = stdout;
        FILE *fp_ips = stdout;
        FILE *fp_csv = NULL;
        int i;
        int port = 23;
```

T-Pot


Home
Kibana
ES Head Plugin
UI-For-Docker
WebSSH
Netdata
LIFE IS FOR SHARING.
Discover
Visualize
Dashboard
Settings
Last 15 minut
Default
*
Welcome to T-Pot
Welcome to your shiny new T-Pot 16.10 installation!
Before you get started tell Kibana what installation type you have chosen for T-Pot.
You can now click the Load Saved Dashboard button in the upper right corner to load your desired dashboard.
Make sure to click the Save Dashboard button and save your dashboard as Default.
If you do not want to see this reminder any longer, just click on the (X) in the upper right corner of this visualization and save the dashboard one more time.
Cowrie - E...
0
Events
Dionaea - ...
3
Events
ElasticPot ...
0
Events
Glastopf - ...
4
Events
Honeytrap ...
6
Events
Suricata - ...
253
Count

T-Pot (after nmap -sS -sV)



Awesome Honeypots

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

Artillery

- Installation manual: https://www.binarydefense.com/files/Artillery_Installation_Manual.pdf
- https://www.binarydefense.com/project-artillery/

Artillery

```
main@main-VirtualBox:~/artillery$ sudo python setup.py

Welcome to the Artillery installer. Artillery is a honeypot, file monitoring, and overall security tool used to protect your nix systems.

Written by: Dave Kennedy (ReL1K)

Do you want to install Artillery and have it automatically run when you restart [y/n]: y
[*] Beginning installation. This should only take a moment.
[*] Adding artillery into startup through init scripts..
[*] Triggering update-rc.d on artillery to automatic start...
Do you want to keep Artillery updated? (requires internet) [y/n]: y
[*] Checking out Artillery through github to /var/artillery
Cloning into '/var/artillery'...
remote: Counting objects: 876, done.
remote: Total 876 (delta 0), reused 0 (delta 0), pack-reused 876
Receiving objects: 100% (876/876), 207.83 KiB | 258.00 KiB/s, done.
Resolving deltas: 100% (568/568), done.
Checking connectivity... done.
[*] Finished. If you want to update Artillery go to /var/artillery and type 'git pull'
Would you like to start Artillery now? [y/n]: y
Starting Artillery...                                  Ok
[*] Installation complete. Edit /var/artillery/config in order to config artillery to your liking..
```

Conclusion

- Follow up blog post with links will be at elliottbrink.com
- Slides available to BSides Canberra & above
- Have fun with honeypots!
- Thank you!