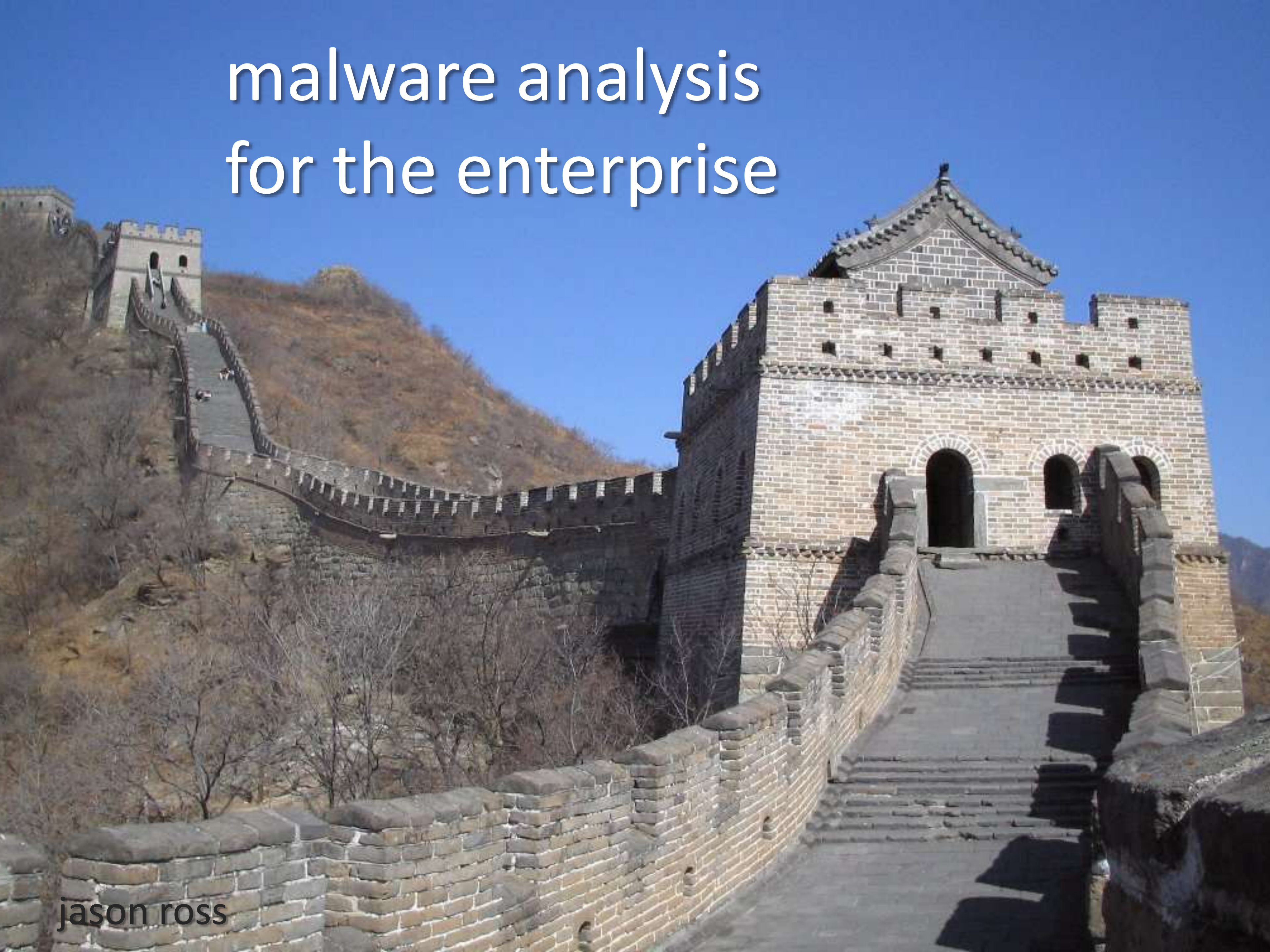
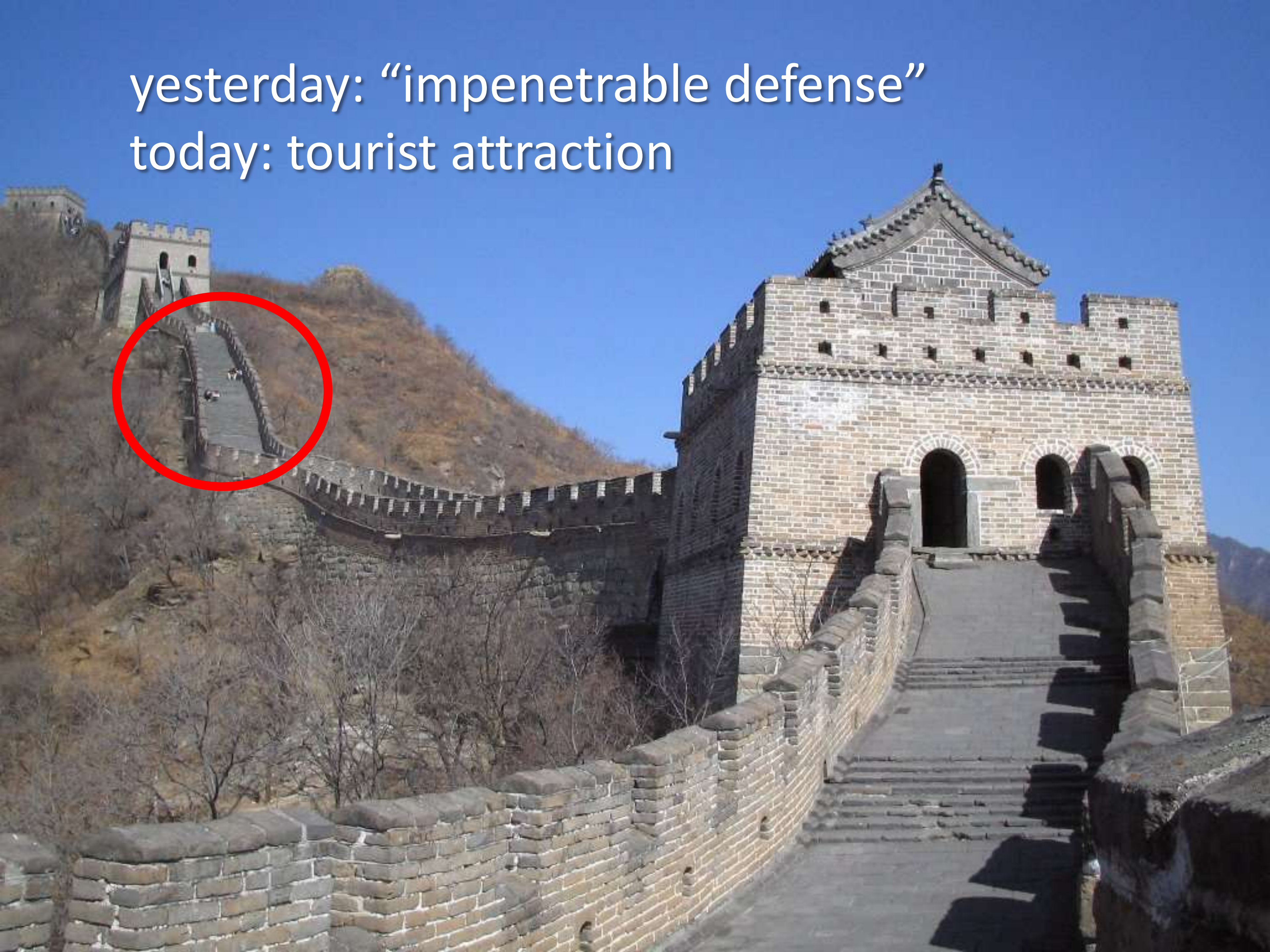
malware analysis for the enterprise



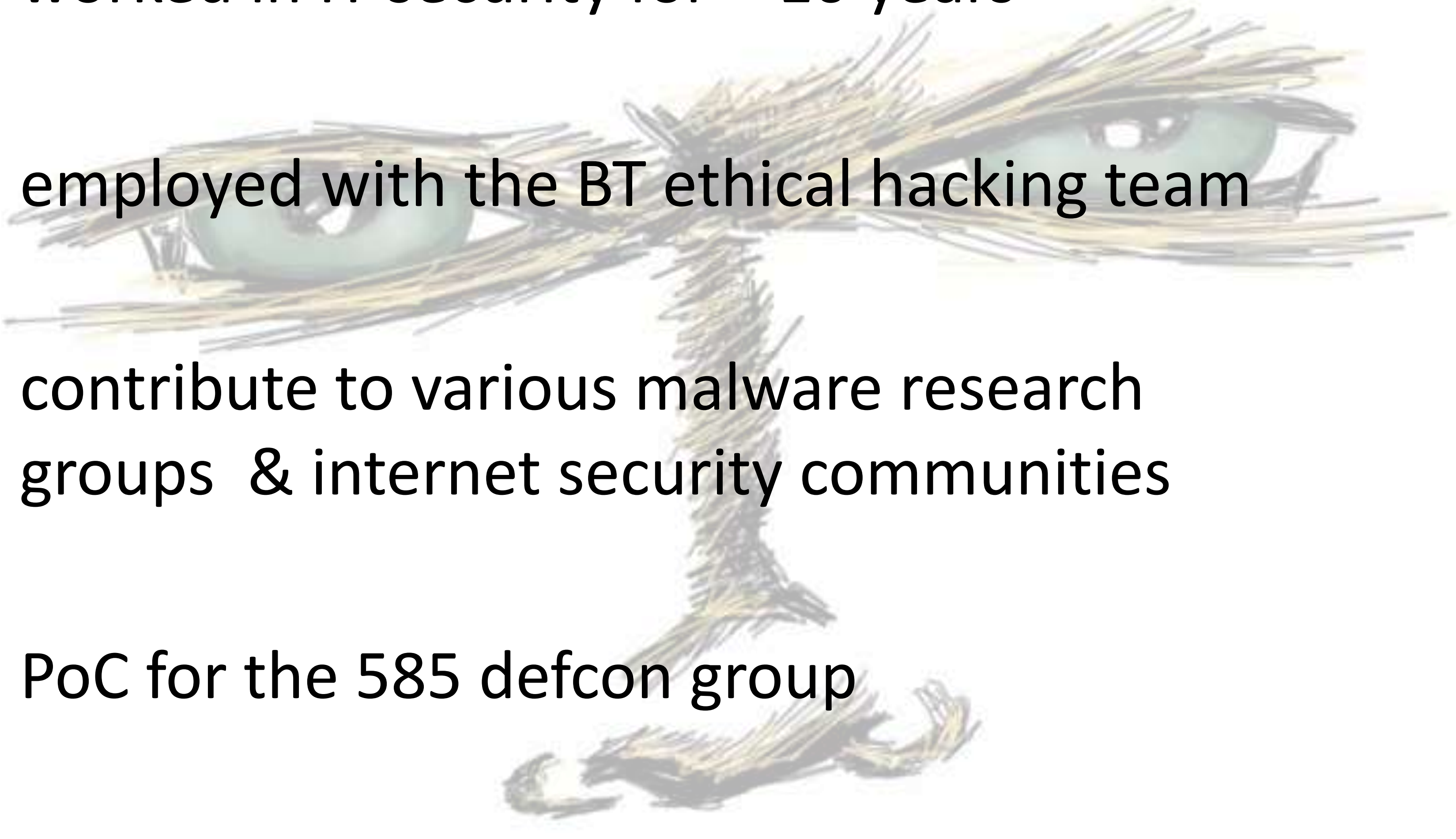
jason ross

yesterday: “impenetrable defense”
today: tourist attraction



obligatory narcissism

- worked in IT security for > 10 years
- employed with the BT ethical hacking team
- contribute to various malware research groups & internet security communities
- PoC for the 585 defcon group



what are we talking about?

we are awesome at compliance!

so why are we getting owned?

because the current security mindset sucks.

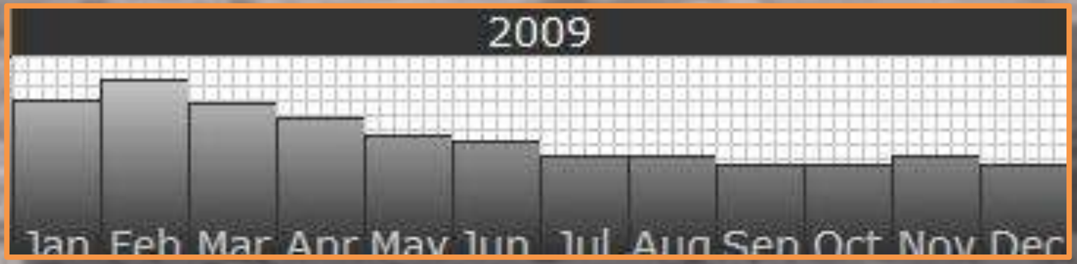
- do you know when a host is compromised?
- can you tell if other hosts were?
- what data was taken?
- do you have any idea where it went?

compliance != security

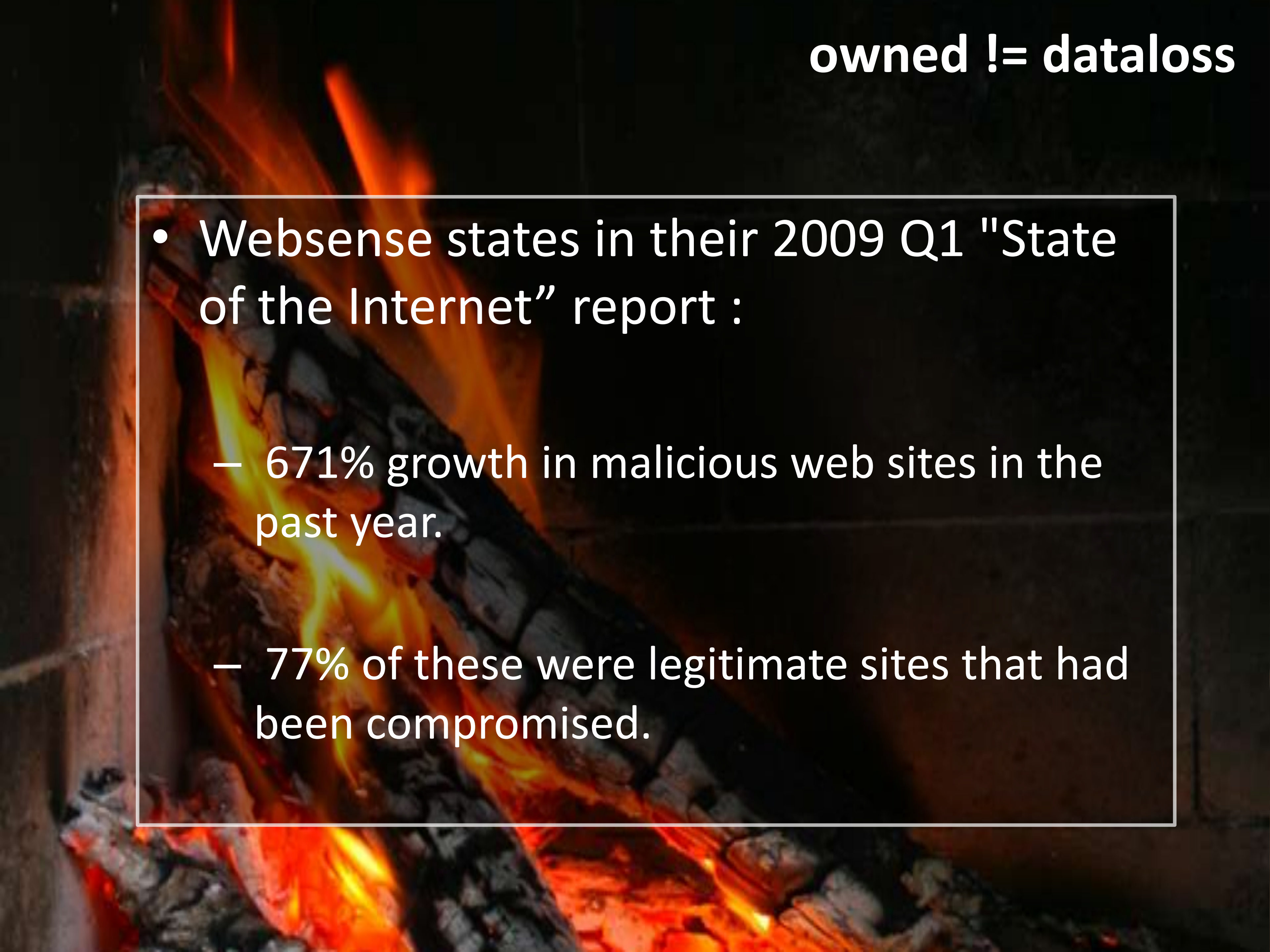
- we’re not getting better at securing systems

- we *are* becoming adept at evading the average security auditor.

"has data loss jumped the shark?"



Despite the statements in the prior slide,
we are seeing a decrease in data loss incidents



owned != dataloss

- Websense states in their 2009 Q1 "State of the Internet” report :
  - 671% growth in malicious web sites in the past year.
  - 77% of these were legitimate sites that had been compromised.

what does that mean?

attackers may not be interested in your data at all.

they may be looking to use your brand image.

the intended victim may not even be your customer.

arms race ++

industry:

let’s use signatures!

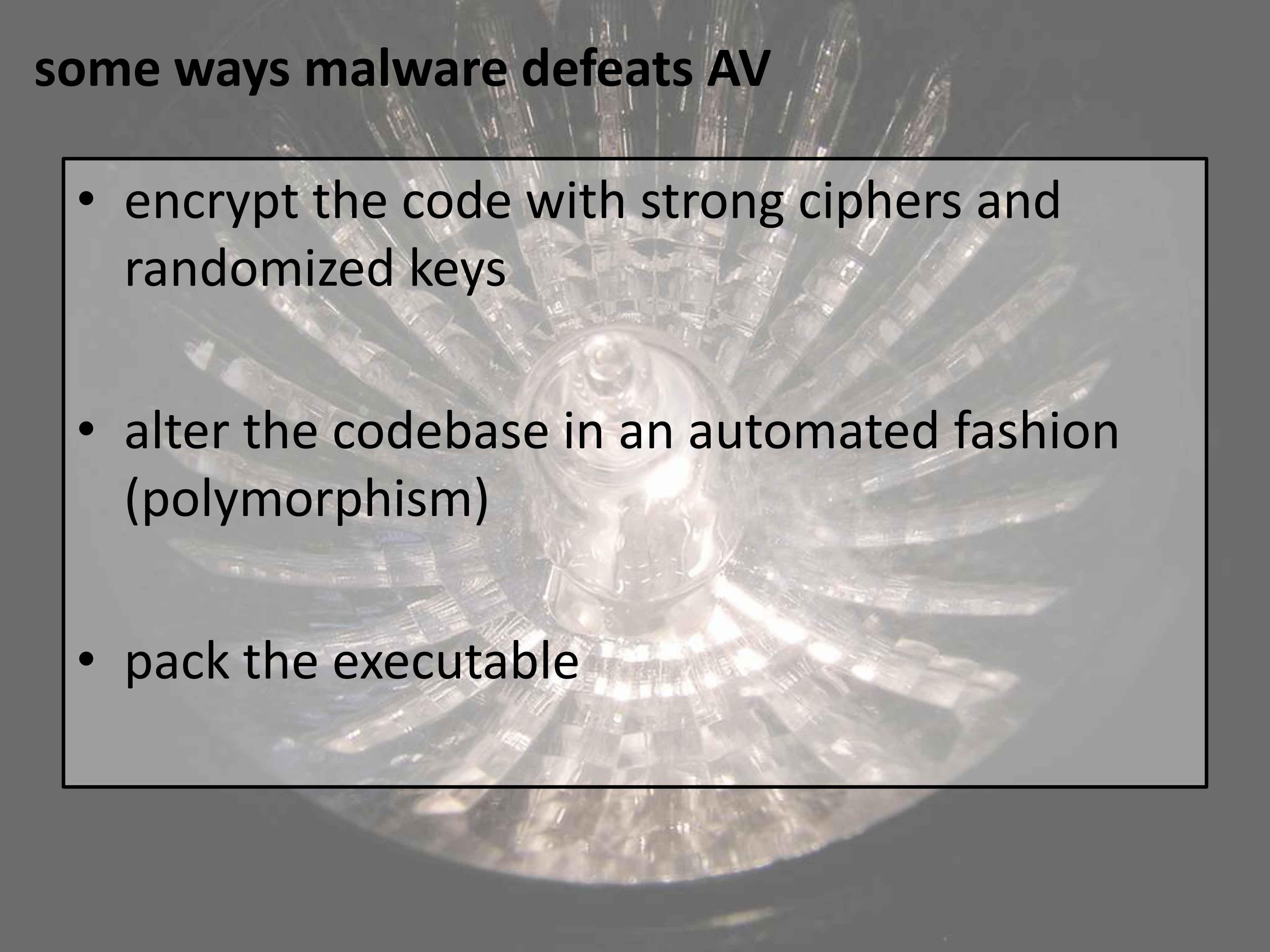
malware:

i can has packing, crypto, and some polymorphism?

protection fail.

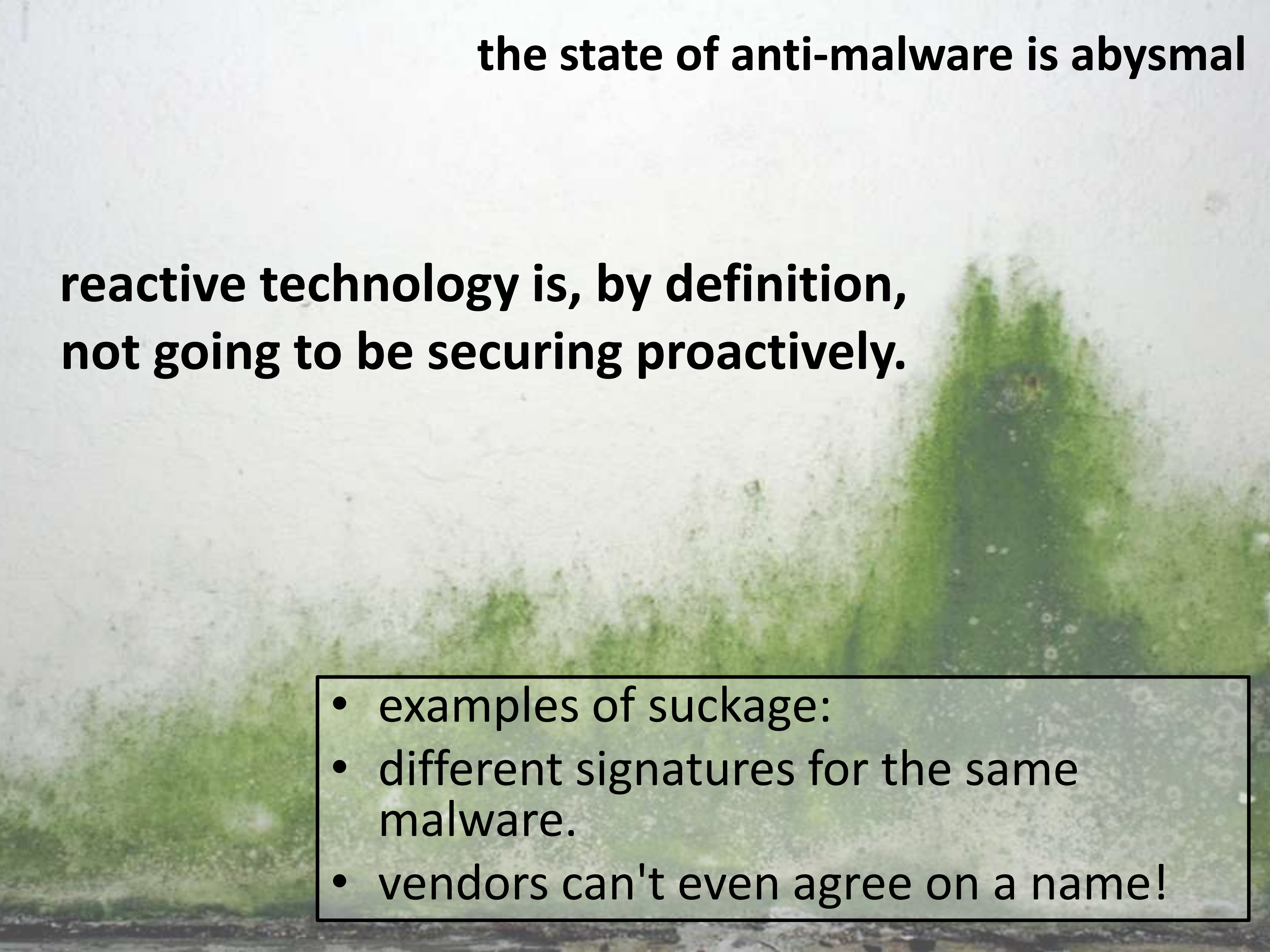
heuristics won't save you

- they can be useful and effective
- they miss things
- especially if multiple stages are involved



some ways malware defeats AV

- encrypt the code with strong ciphers and randomized keys
- alter the codebase in an automated fashion (polymorphism)
- pack the executable



the state of anti-malware is abysmal

reactive technology is, by definition, not going to be securing proactively.

- examples of suckage:
- different signatures for the same malware.
- vendors can't even agree on a name!

but…

my
AV
suite
alerts
me
constantly!

think about that for a minute…

at least AV is catching stuff.
that's good, right?

the host was probably compromised before AV caught whatever it alerted on.

that’s because malware does not infect a host using a single stage process.

prepare for HolyCrap!

how malware works (really!)

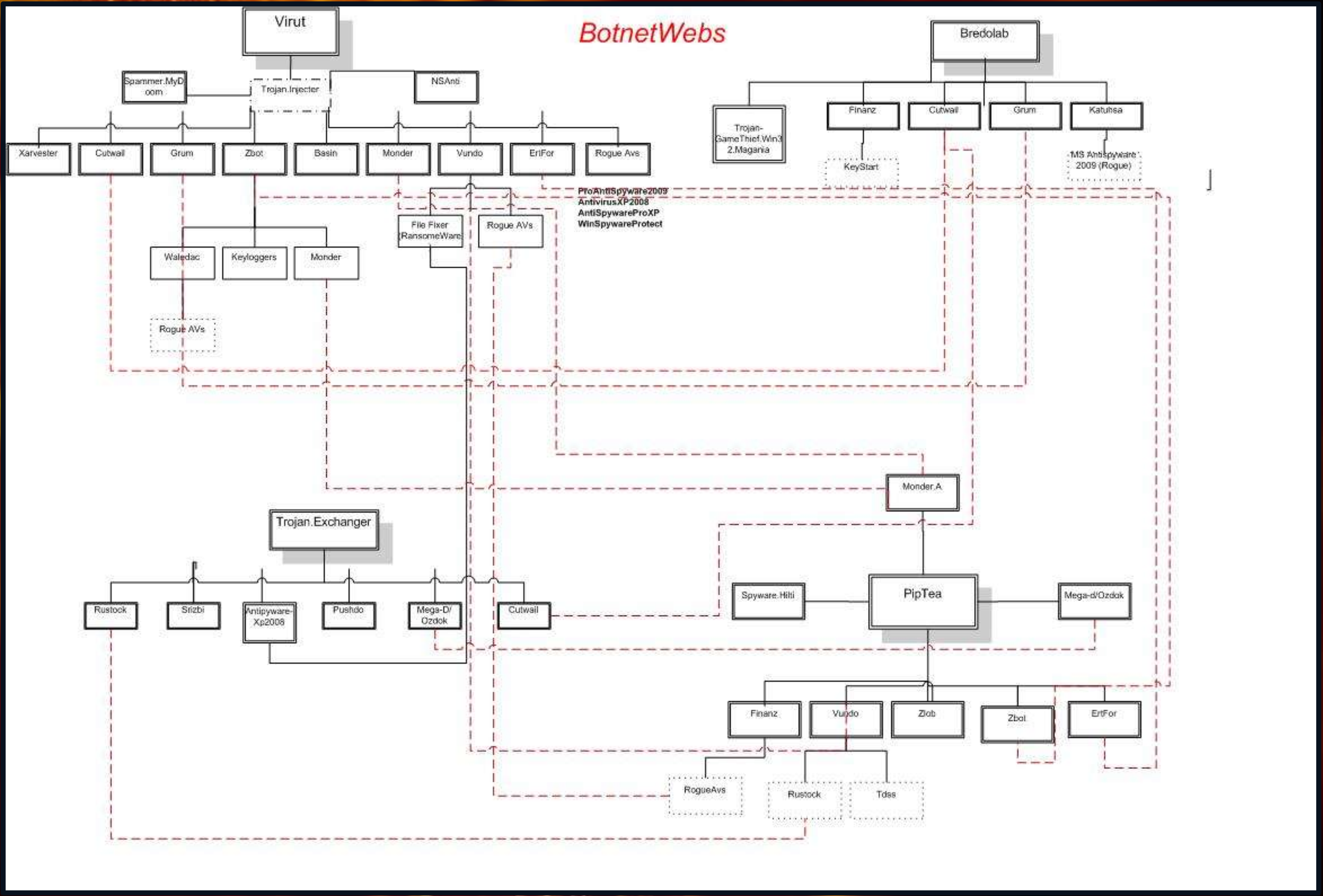
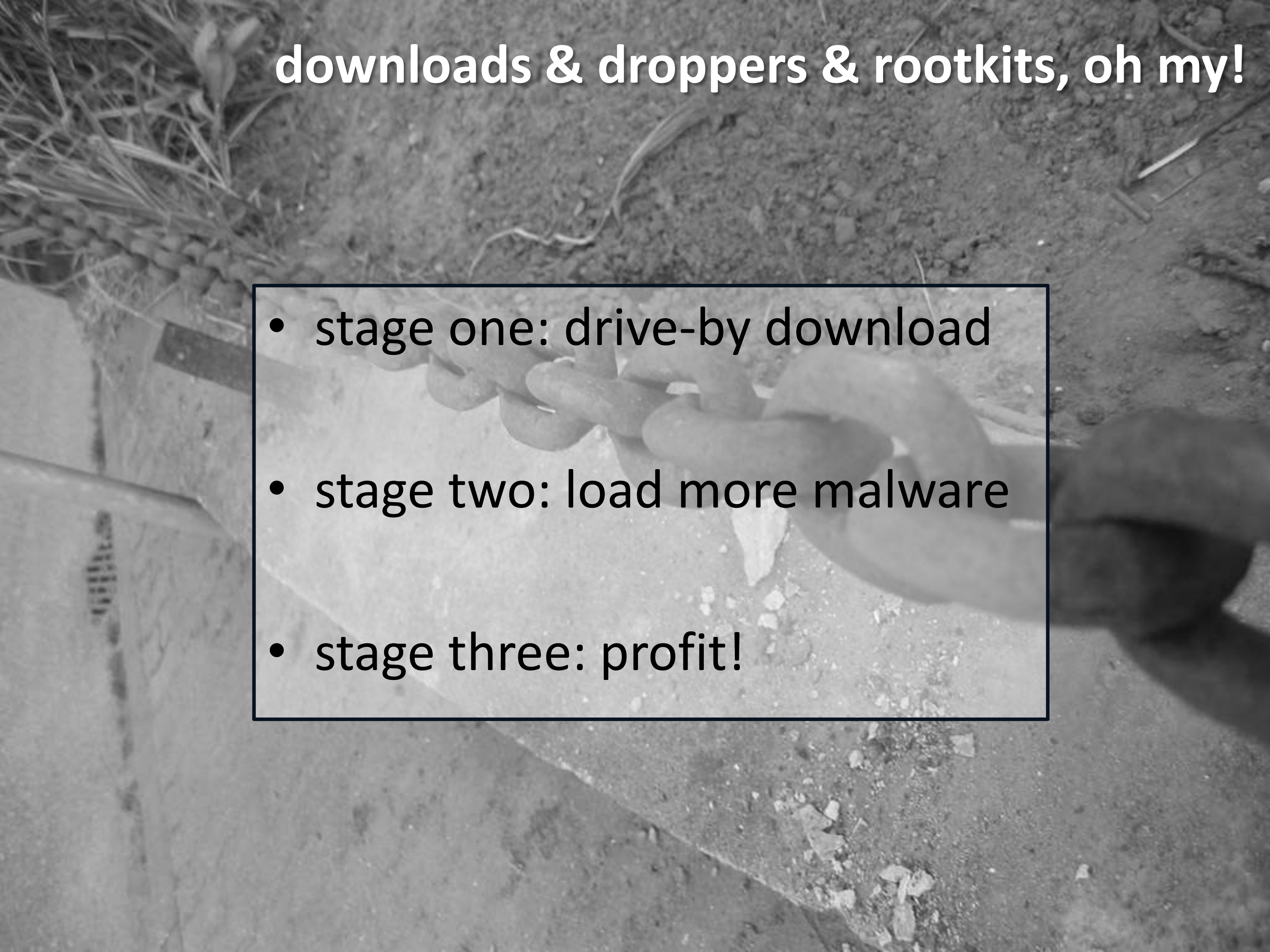


image taken from the FireEye Research Blog:
http://blog.fireeye.com/research/2009/04/botnetweb.html

it's a business, not a kiddie

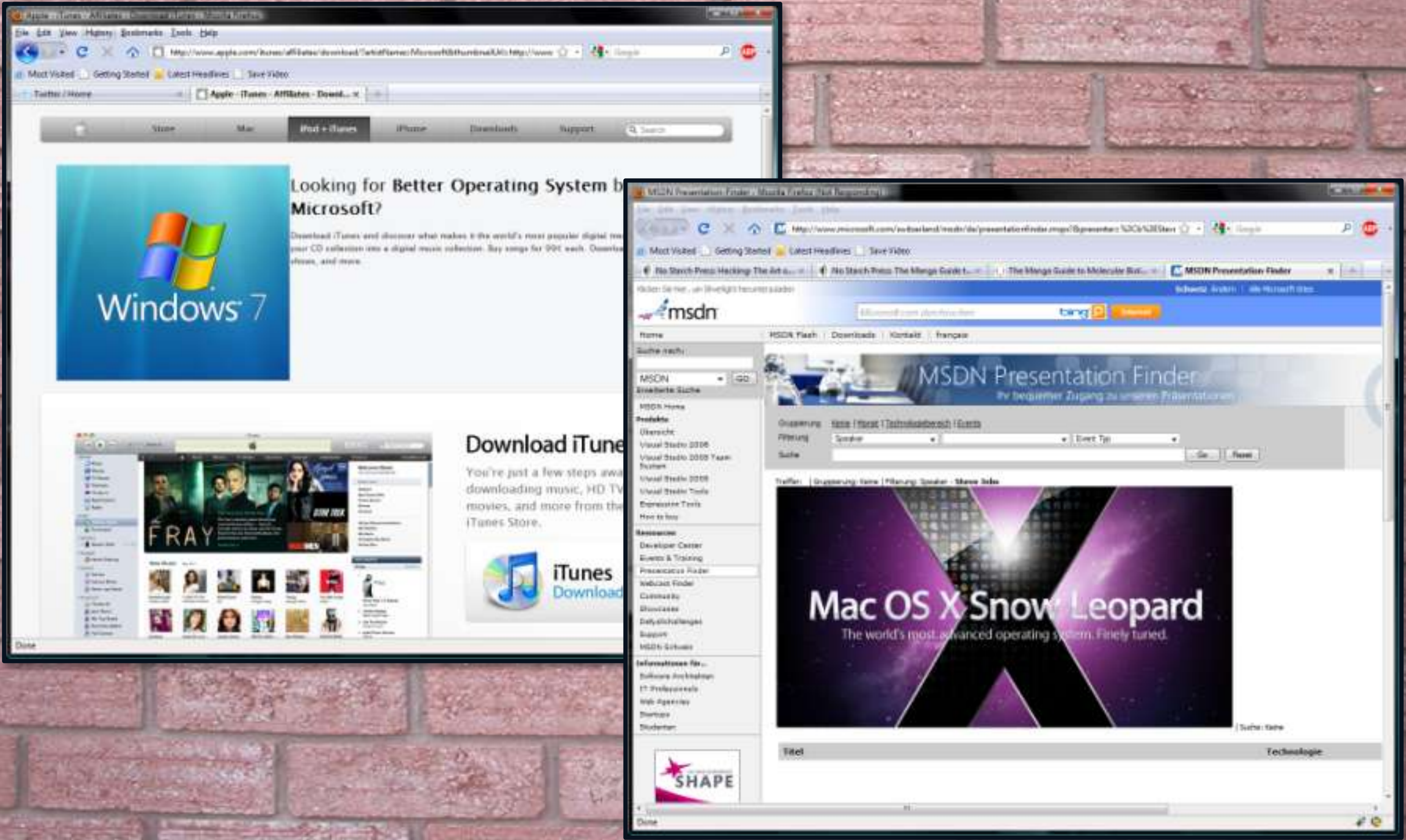
- payroll
- support models
- distribution channels
- strategic partnerships



downloads & droppers & rootkits, oh my!

- stage one: drive-by download
- stage two: load more malware
- stage three: profit!

pfft. only mom & pop sites are being used for droppers, right?



Looking for **Better Operating System** b

Microsoft?

Download iTune

You're just a few steps awa
downloading music, HD TV
movies, and more from the
iTunes Store.

um. wow. that sucks.

- yes. yes it does.
- still think you’re safe because IDS, AV, or even a QSA says so?
- that’s OK, so did these guys:

Heartland PAYMENT SYSTEMS
The Highest Standards | The Most Trusted Transactions

TJX

CARD SYSTEMS
INCORPORATED

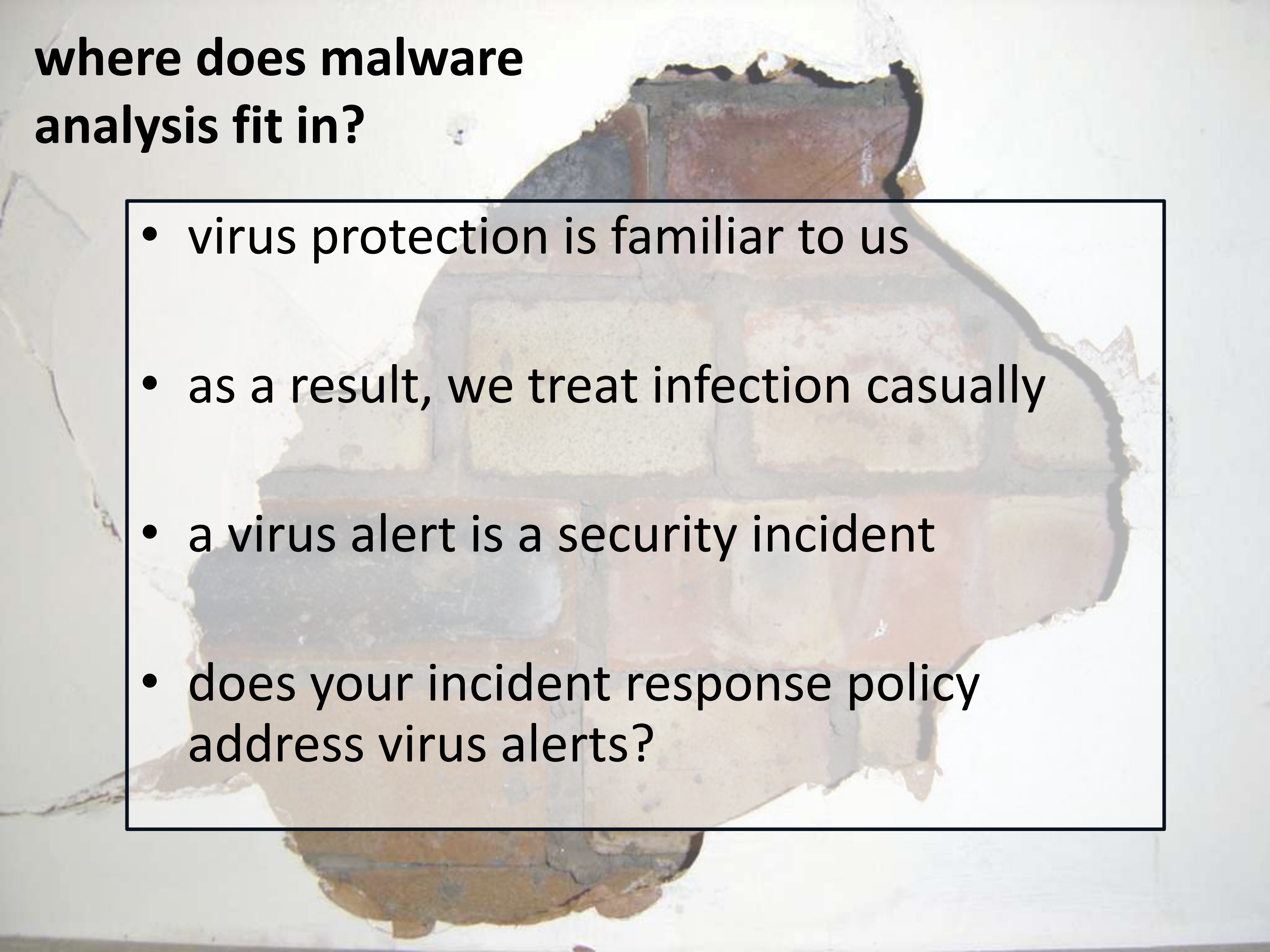
current (2009) US population	:	307 million
records lost by these companies	:	264 million
percentage of population “owned”	:	~86 %

more lessons from heartland

- malcode authors are invested in long term solutions
- malware is increasingly targeted

scary example time

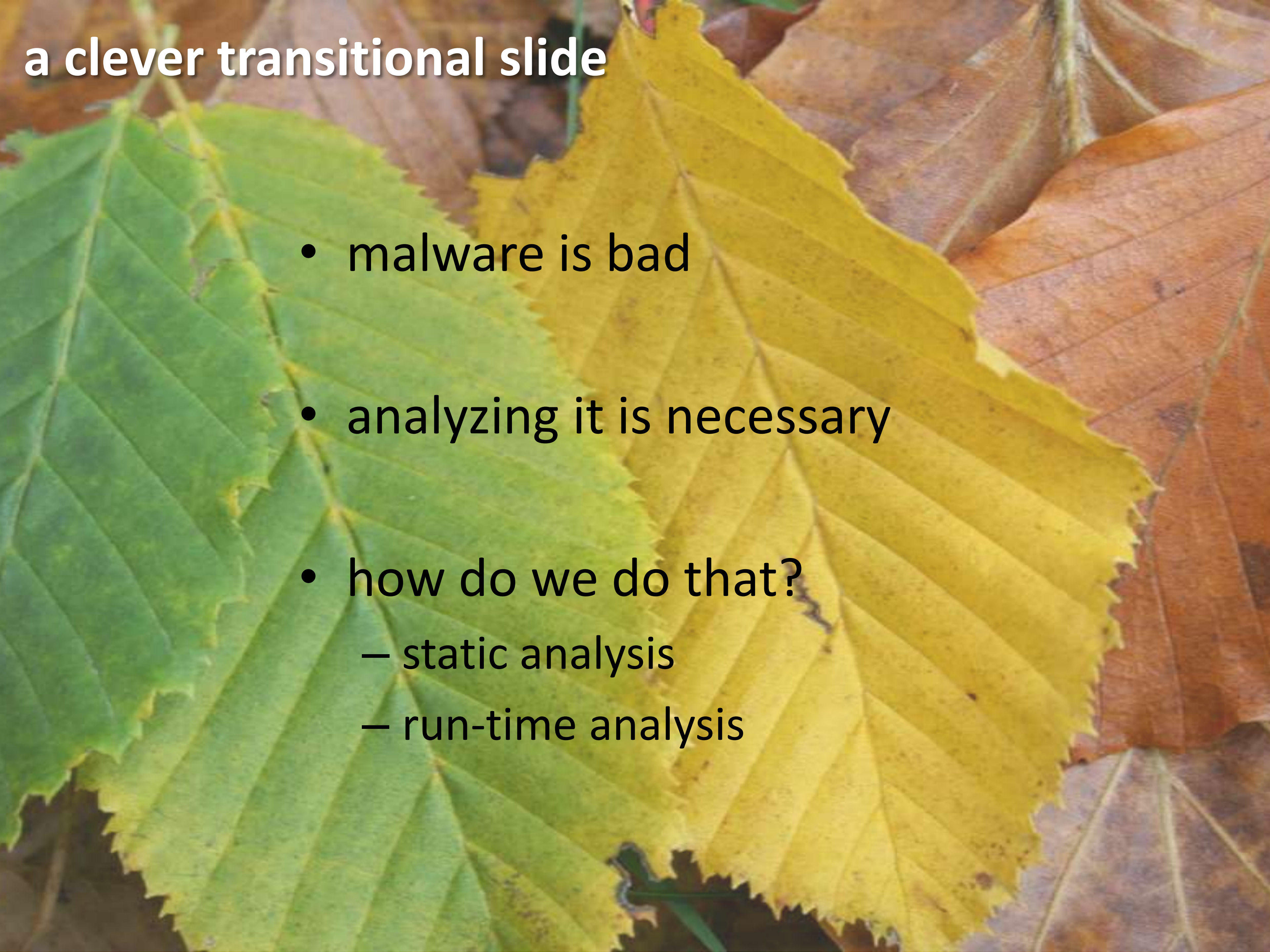
- URLZone
  - My balance is fine!
- Monkif / DIKhora
  - Nothing here but us JPEGs



where does malware analysis fit in?

- virus protection is familiar to us
- as a result, we treat infection casually
- a virus alert is a security incident
- does your incident response policy address virus alerts?

a clever transitional slide



- malware is bad
- analyzing it is necessary
- how do we do that?
  - static analysis
  - run-time analysis

sandnets

playing with
fire is cool!

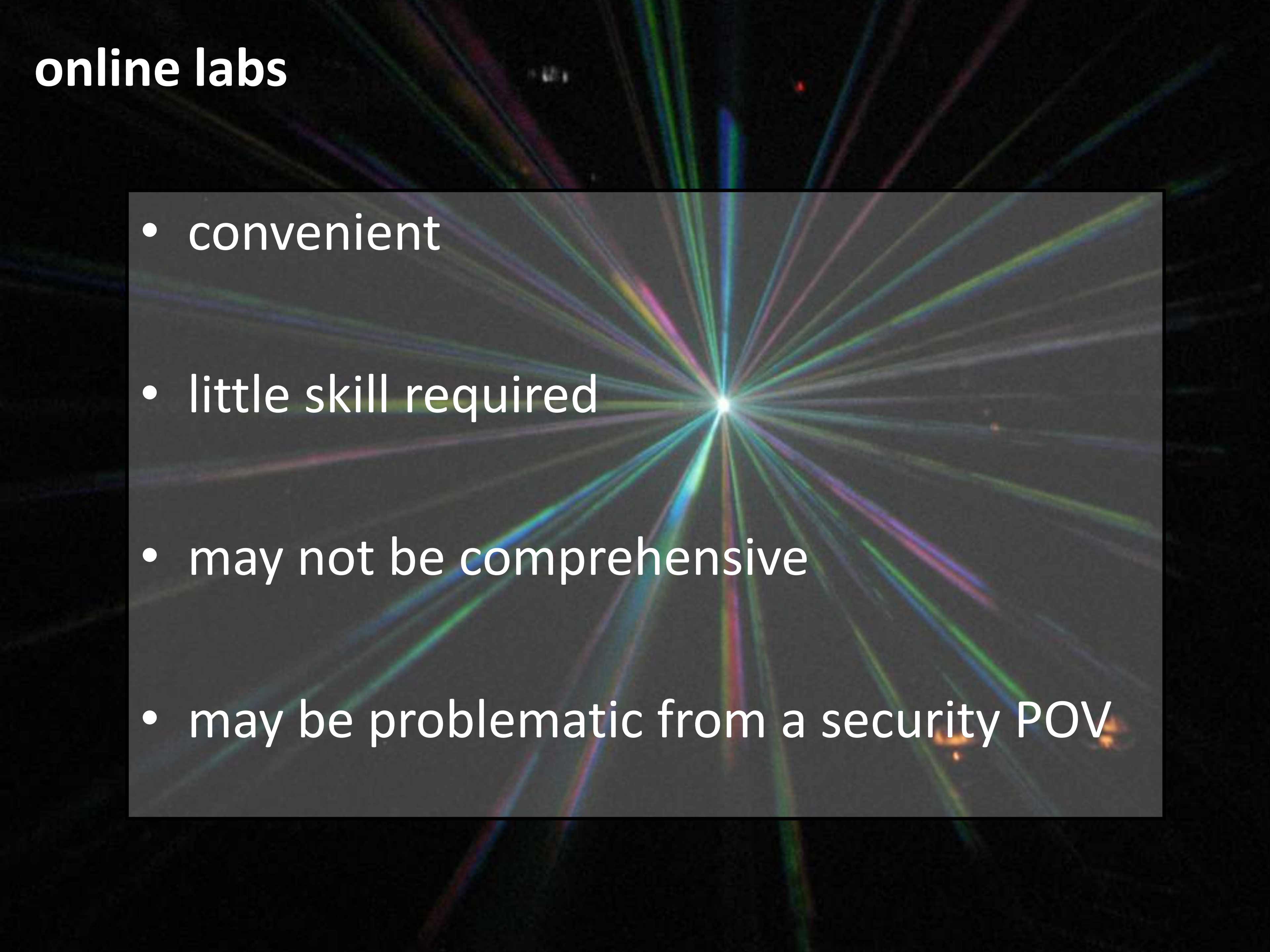
what is a sandnet?

- a test environment using multiple hosts
- isolated from the production network
- used to analyze malicious software

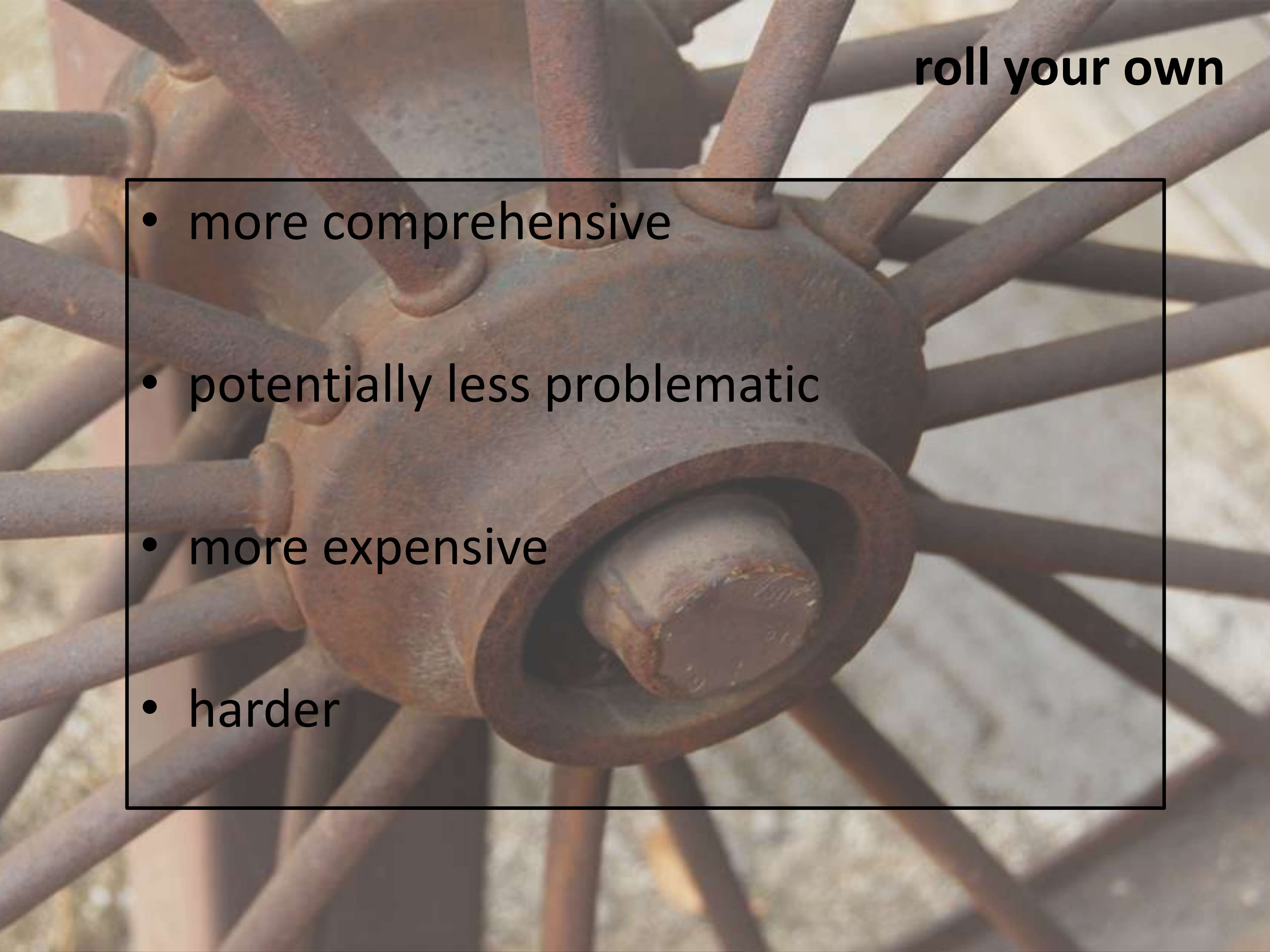
what are the options?

- online labs
- virtual machines
- bare metal

online labs

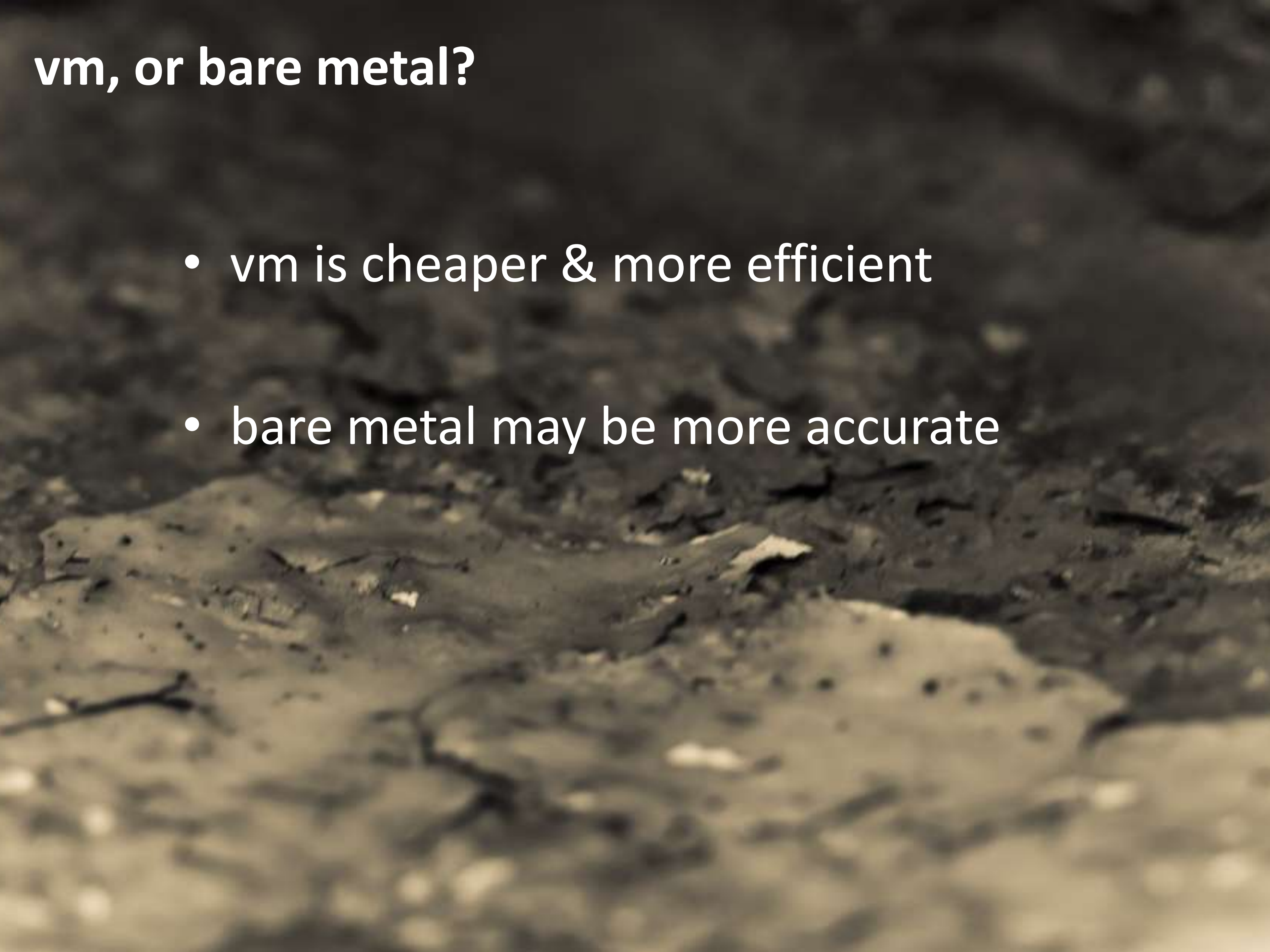


- convenient
- little skill required
- may not be comprehensive
- may be problematic from a security POV



roll your own

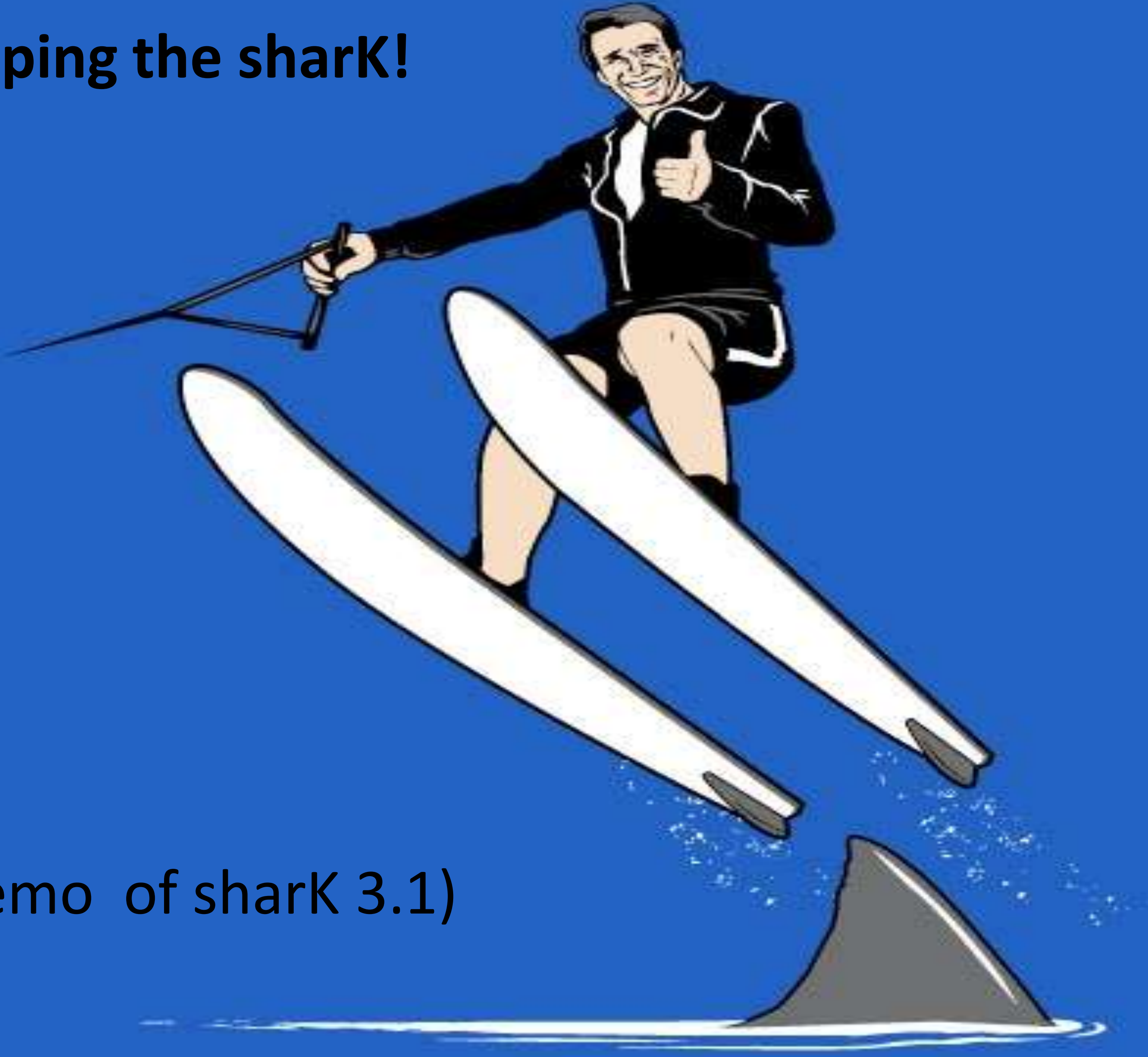
- more comprehensive
- potentially less problematic
- more expensive
- harder



vm, or bare metal?

- vm is cheaper & more efficient
- bare metal may be more accurate

jumping the sharK!



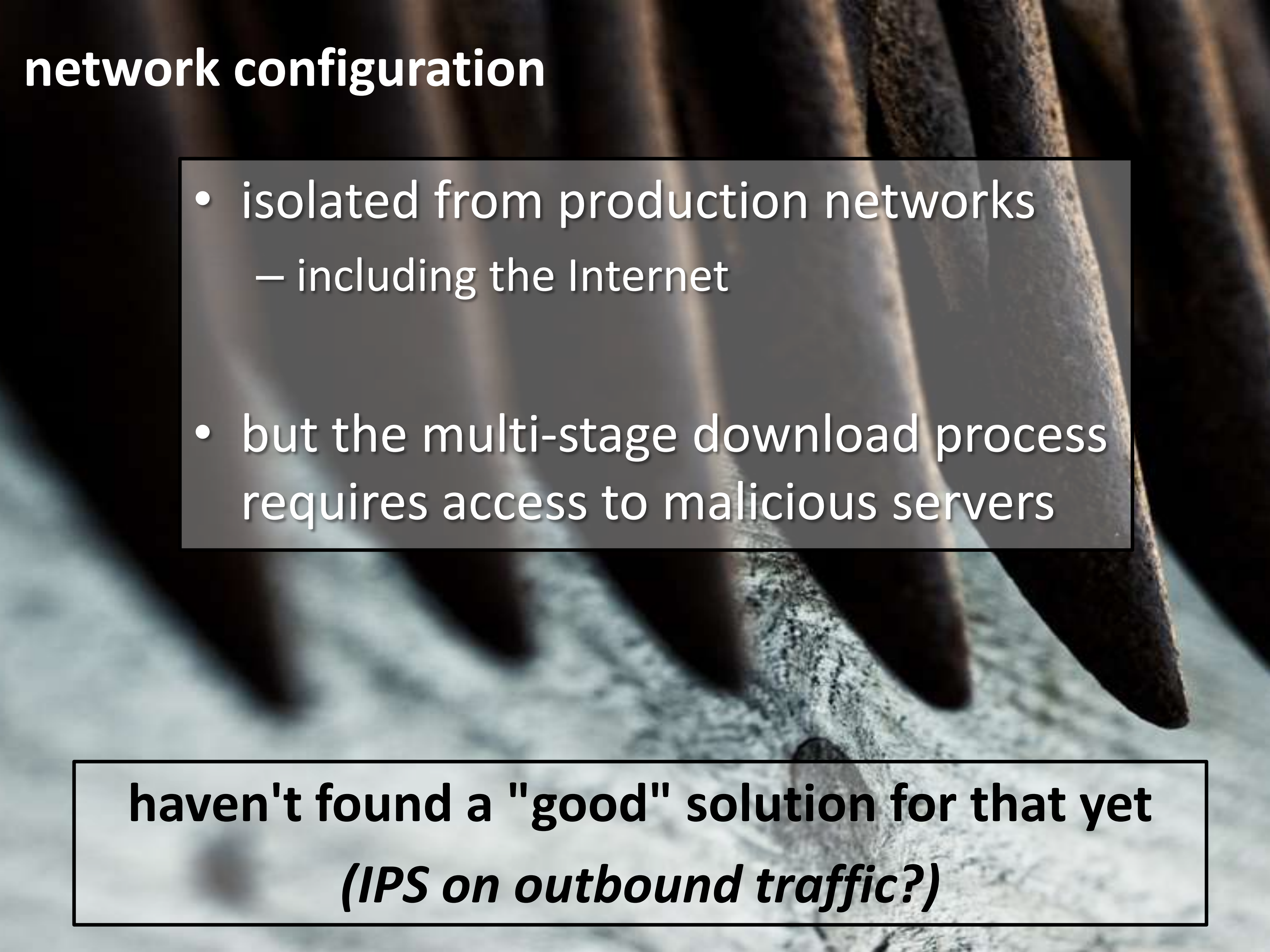
(demo of sharK 3.1)

how many hosts?

- At least 2 probably
  - Victim
  - Services / Monitoring

```
> VBoxManage list vms
"linux" {ad59f194-585e-49c5-a54c-5e92322b1188}
"winxp_sp3_01" {7a554f4e-6aea-42f1-a3c5-488d43f161ff}
```

network configuration

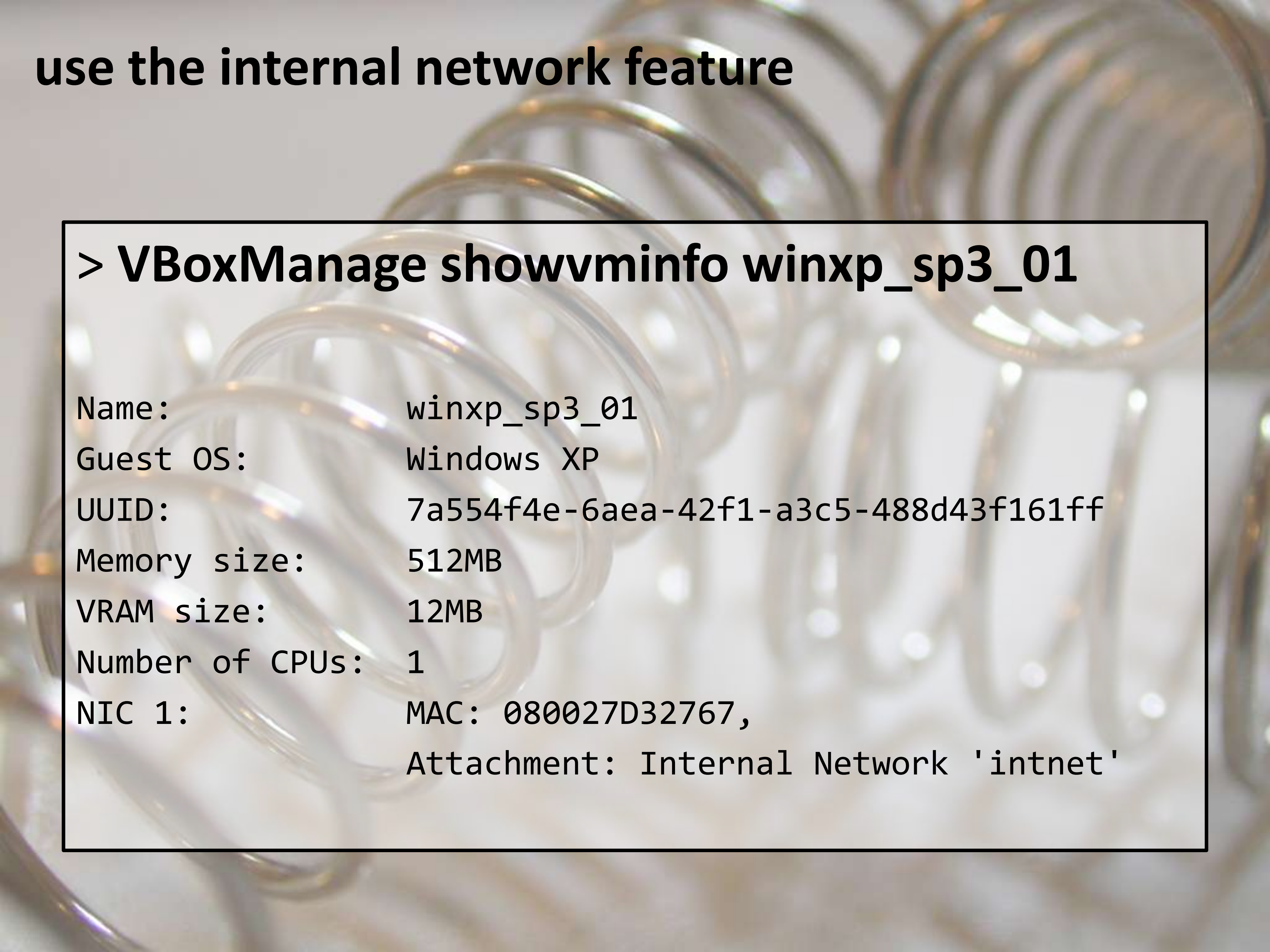


- isolated from production networks
  - including the Internet
- but the multi-stage download process requires access to malicious servers

haven't found a "good" solution for that yet

(IPS on outbound traffic?)

use the internal network feature



```
> VBoxManage showvminfo winxp_sp3_01

Name:              winxp_sp3_01
Guest OS:          Windows XP
UUID:              7a554f4e-6aea-42f1-a3c5-488d43f161ff
Memory size:       512MB
VRAM size:         12MB
Number of CPUs:    1
NIC 1:             MAC: 080027D32767,
                   Attachment: Internal Network 'intnet'
```

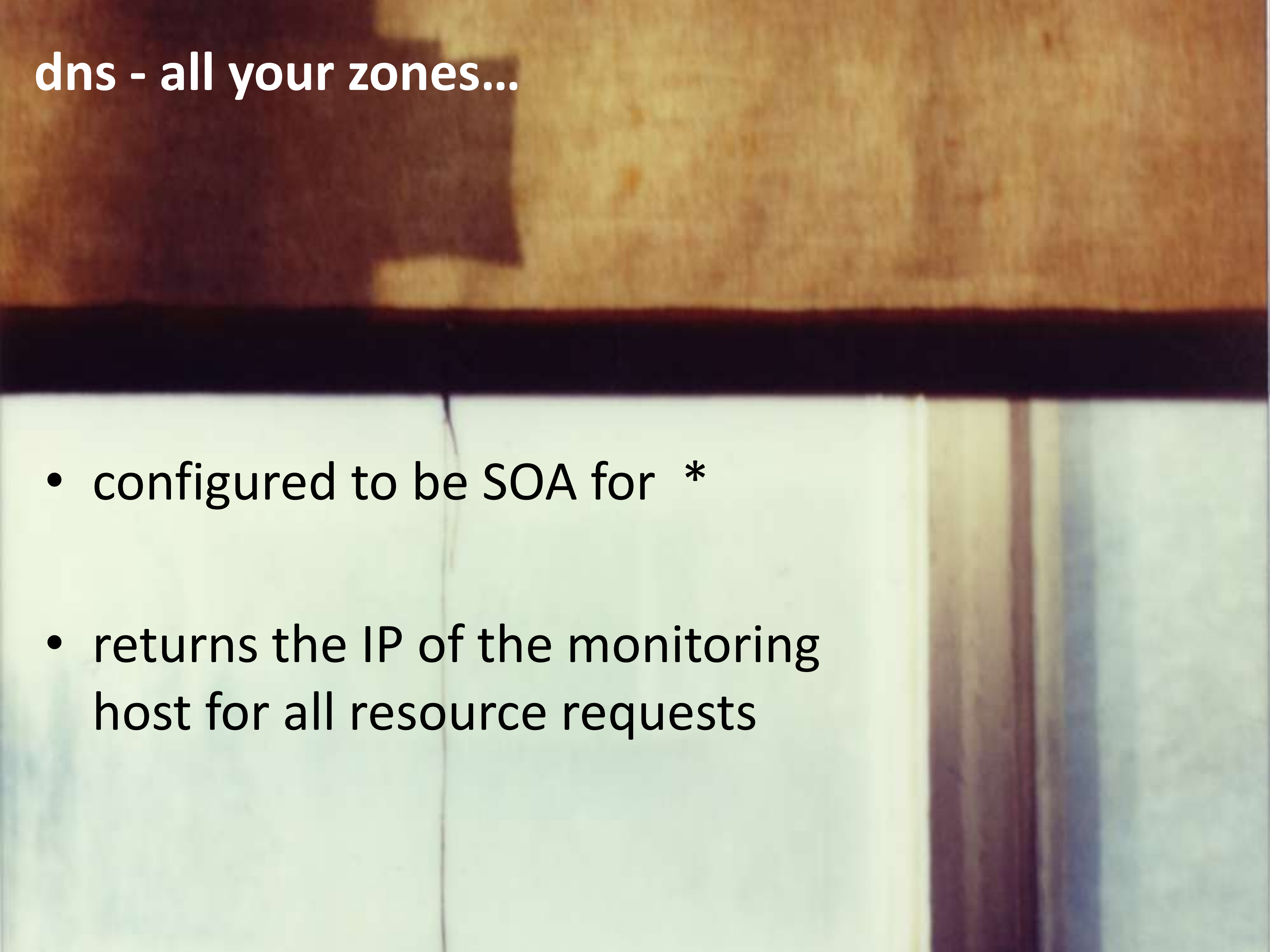
dhcp - because dynamic is easy

```
> VBoxManage dhcpserver add
--netname intnet
--ip 192.168.3.1
--netmask 255.255.255.0
--lowerip 192.168.3.100
--upperip 192.168.3.250
--enable
```

monitoring traffic

- let the VM do the work for you

```
> VBoxManage modifyvm linux –nictrace1
on –nictracefile1 “C:\Users\Test\linux.pcap”
```



dns - all your zones…

- configured to be SOA for *
- returns the IP of the monitoring host for all resource requests

Remember the MX

db.wildcard

```
$TTL   604800
@   IN   SOA   localhost.  root.localhost. (
                         2010012201  ; serial
                             604800  ; refresh
                              86400  ; retry
                            2419200  ; expire
                             604800) ; negative
  cache ttl

@              IN          NS         localhost.
*              IN          MX 10      192.168.3.101
*              IN           A         192.168.3.101
```

apache configuration

- mod_forensic is your friend

- configuration is easy:

```
ForensicLog    /var/log/apache2/forensic_log
```

- Enable and reload:

```
# a2enmod log_forensic
# apache2ctl reload
```

sample forensic log

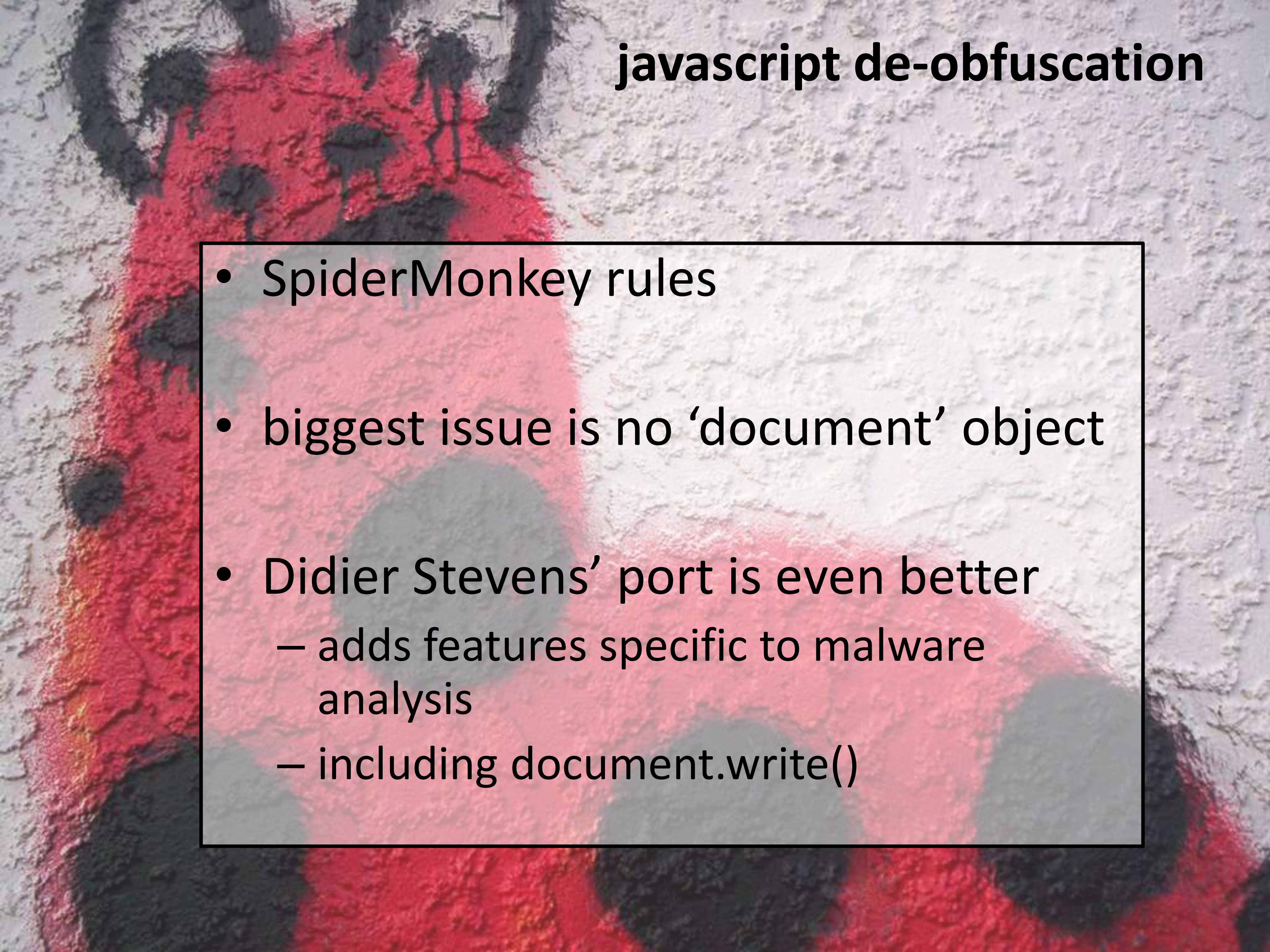
```
+2021:4adf8568:0
  |GET / HTTP/1.1
  |Accept:*/*
  |Accept-Language:en-us
  |Accept-Encoding:gzip, deflate
  |User-Agent:Mozilla/4.0
     (compatible; MSIE 6.0;
      Windows NT 5.1; SV1)
  |Host:192.168.3.101
  |Connection:Keep-Alive
  |Cache-Control:no-cache
-2021:4adf8568:0
```

fun with netcat

- very easy to set up:

# netcat –nvlp 8080 –o tcp_8080.txt

```
< 00000000 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31 0d 0a # GET / HTTP/1.1..
< 00000010 41 63 63 65 70 74 3a 20 69 6d 61 67 65 2f 67 69 # Accept: image/gi
< 00000020 66 2c 20 69 6d 61 67 65 2f 78 2d 78 62 69 74 6d # f, image/x-xbitm
< 00000030 61 70 2c 20 69 6d 61 67 65 2f 6a 70 65 67 2c 20 # ap, image/jpeg,
< 00000040 69 6d 61 67 65 2f 70 6a 70 65 67 2c 20 61 70 70 # image/pjpeg, app
< 00000050 6c 69 63 61 74 69 6f 6e 2f 78 2d 73 68 6f 63 6b # lication/x-shock
< 00000060 77 61 76 65 2d 66 6c 61 73 68 2c 20 2a 2f 2a 0d # wave-flash, */*.
< 00000070 0a 41 63 63 65 70 74 2d 4c 61 6e 67 75 61 67 65 # .Accept-Language
< 00000080 3a 20 65 6e 2d 75 73 0d 0a 41 63 63 65 70 74 2d # : en-us..Accept-
< 00000090 45 6e 63 6f 64 69 6e 67 3a 20 67 7a 69 70 2c 20 # Encoding: gzip,
< 000000a0 64 65 66 6c 61 74 65 0d 0a 55 73 65 72 2d 41 67 # deflate..User-Ag
< 000000b0 65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 34 2e 30 # ent: Mozilla/4.0
< 000000c0 20 28 63 6f 6d 70 61 74 69 62 6c 65 3b 20 4d 53 #  (compatible; MS
< 000000d0 49 45 20 36 2e 30 3b 20 57 69 6e 64 6f 77 73 20 # IE 6.0; Windows
< 000000e0 4e 54 20 35 2e 31 3b 20 53 56 31 29 0d 0a 48 6f # NT 5.1; SV1)..Ho
< 000000f0 73 74 3a 20 31 39 32 2e 31 36 38 2e 33 2e 31 30 # st: 192.168.3.10
< 00000100 31 3a 38 30 38 30 0d 0a 43 6f 6e 6e 65 63 74 69 # 1:8080..Connecti
< 00000110 6f 6e 3a 20 4b 65 65 70 2d 41 6c 69 76 65 0d 0a # on: Keep-Alive..
< 00000120 0d 0a                                           # ..
```



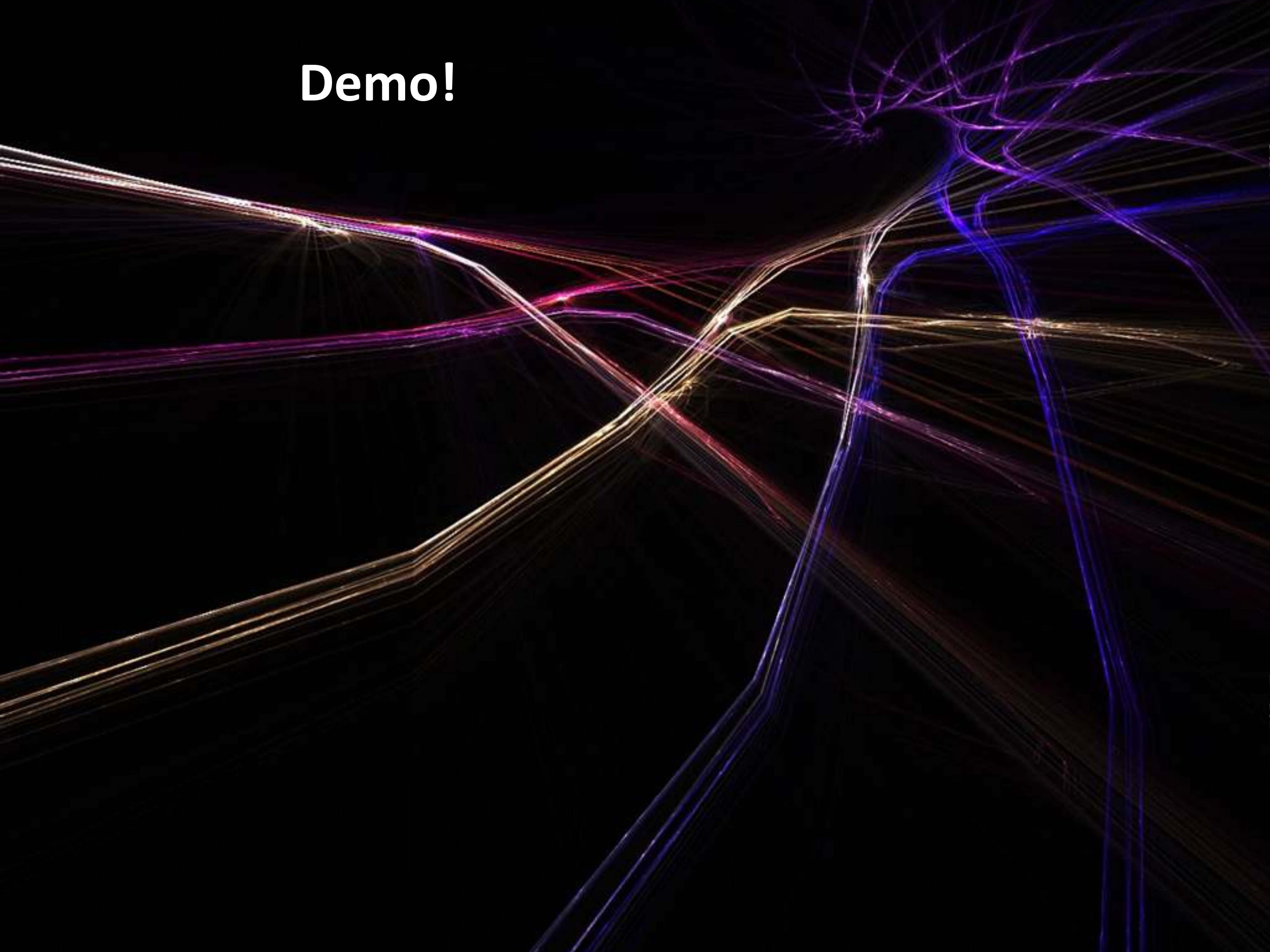
javascript de-obfuscation

- SpiderMonkey rules
- biggest issue is no ‘document’ object
- Didier Stevens’ port is even better
  - adds features specific to malware analysis
  - including document.write()

victim host

- iDefense malcode analyst pack
- Regshot
- strings
- wireshark

Demo!



online resources

Anubis:

- http://anubis.iseclab.org/

Virus Total

- http://www.virustotal.com/

CERT.at Do-It-Yourself Kit

- http://cert.at/downloads/papers/mass_malware_analysis_en.html

the end

if you want to contact me for
some crazy reason, here's how you can:

https://twitter.com/rossja

algorythm@gmail.com