Cérebro na Banheira e técnicas de evasão

Felipe Almeida
Lucas Moura

Quem somos nós

- Felipe Almeida

- Lucas de Moura

Contexto

- Malware
- Por que Brain-in-a-vat?
- Análise Dinâmica e Técnicas de evasão
- Público-alvo

Objetivos

- Como funciona análise dinâmica de malwares?

- Por que o uso de análise dinâmica cresce?

- Como funcionam as técnicas para evasão de sandboxes?
  - Como defender-se delas?

DISCLAIMER

- Essa fala não versa sobre mobile malware

- Nem sandbox escape

Brain in a vat

- Como eu sei que não sou um cérebro na banheira recebendo estímulos eletro-eletrônicos?

- Como um malware detecta que está rodando em um ambiente simulado?

Malware - Tipos de Análise

- Estática
  - *strings*
  - Decompile
    - JVM, .NET CLR
  - Assembly
  - ↑ VBScript (*.vbs, .vbe*), etc.
- Dinâmica
  - Análise de comportamento
    - Depois da execução, o que o malware faz?
    - Arquivos abertos, atividade de rede, ...?
- Vantagens e desvantagens?
- **Automatizar!**

Typos de malware

- Trojan
- Vírus
- Worm
- Banker
- Dropper

- Evasion?

- Não faremos distinção

simulation (n.)

mid-14c., "a false show, false profession," from Old French simulation "pretence" and directly from Latin simulationem (nominative simulatio) [...] **Meaning "a model or mock-up for purposes of experiment or training" is from 1954.**

http://www.etymonline.com/index.php?term=simulation

Evasão

- The act of avoiding something that you do not want to do or deal with : the act of evading something (Merriam-Webster)
- Malware
  - Comportamento sensível ao ambiente
- Primeiras técnicas
  - Anti-debug
- O que temos hoje?
  - Anti[anti-vm]
  - Diversas formas
  - Cuckoo

Ambiente

- Oracle VirtualBox
  - http://www.virtualbox.org
- Cuckoo Sandbox [modified]
  - http://www.cuckoosandbox.org
  - https://github.com/brad-accuvant/cuckoo-modified
- Windows XP SP3 [32-bit]

Cuckoo Sandbox

- Análise dinâmica de malwares
  - Automatizada

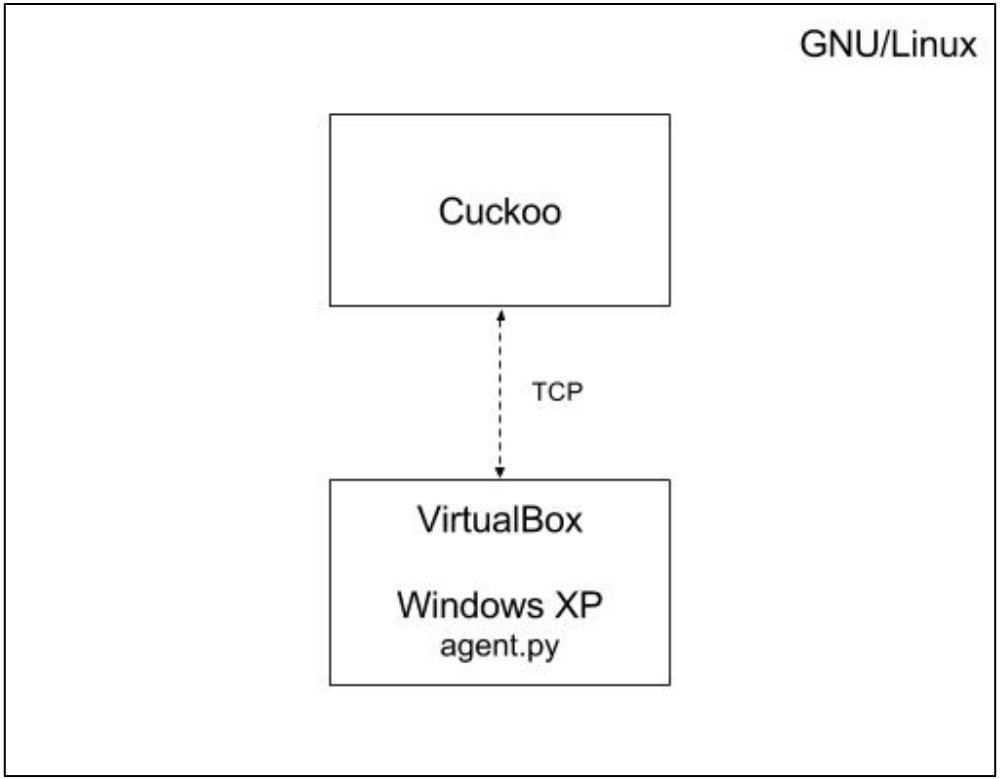
- VirtualBox, VMWare, Xen, etc.

- Roda na VM também
  - cuckoomon.dll
  - agent.py

Cuckoo Sandbox - Relatório

- Chamadas API
- Arquivos
  - Criados, lidos, modificados
  - Dump
- Serviços
  - Criados, iniciados
- Registro
- Comandos executados
- Tráfego de rede
- Screenshots
- *strings*, Virustotal, Volatility, etc.

Cuckoo Sandbox



Cuckoo simplificado. Comunicação com o host.

Técnicas de Evasão (Anti-VM)

E [Anti]Técnicas também

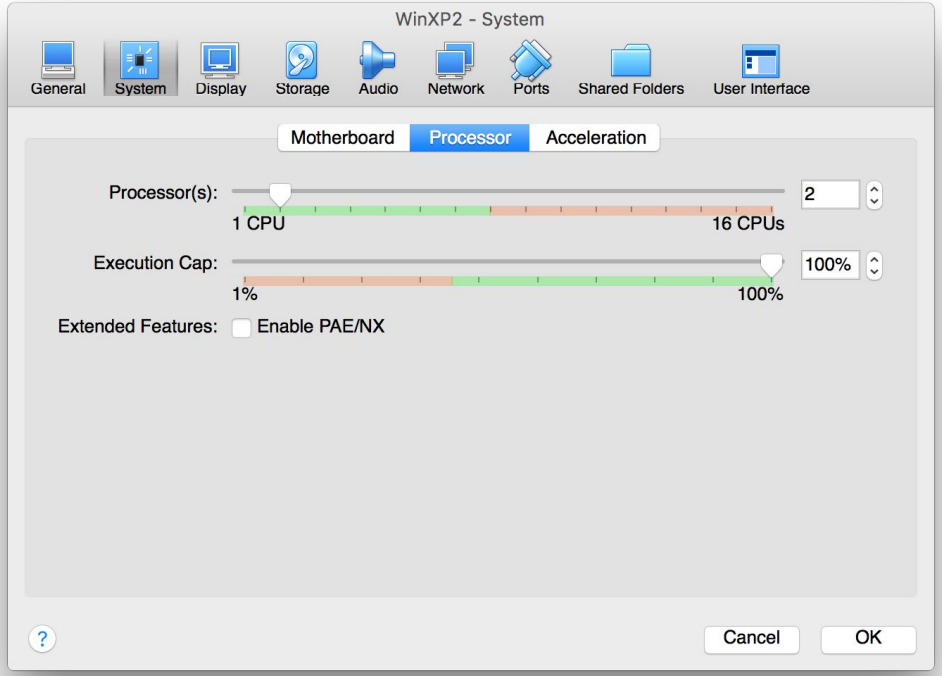
1 - Nro. de processadores

- Utilizada pelo Dyre Wolf em maio de 2015
  - http://www.seculert.com/blogs/new-dyre-version-yet-another-malware-evading-sandboxes

```
int main() {
    SYSTEM_INFO sysinfo;
    GetSystemInfo(&sysinfo);

    int nro_CPU = sysinfo.dwNumberOfProcessors;
    if (nro_CPU == 1) {
      vazei_fui_adios();
      return 0;
     }
    roda_malware_rouba_tudo();
    return 0;
}
```

[Anti] 1 - Nro. de processadores



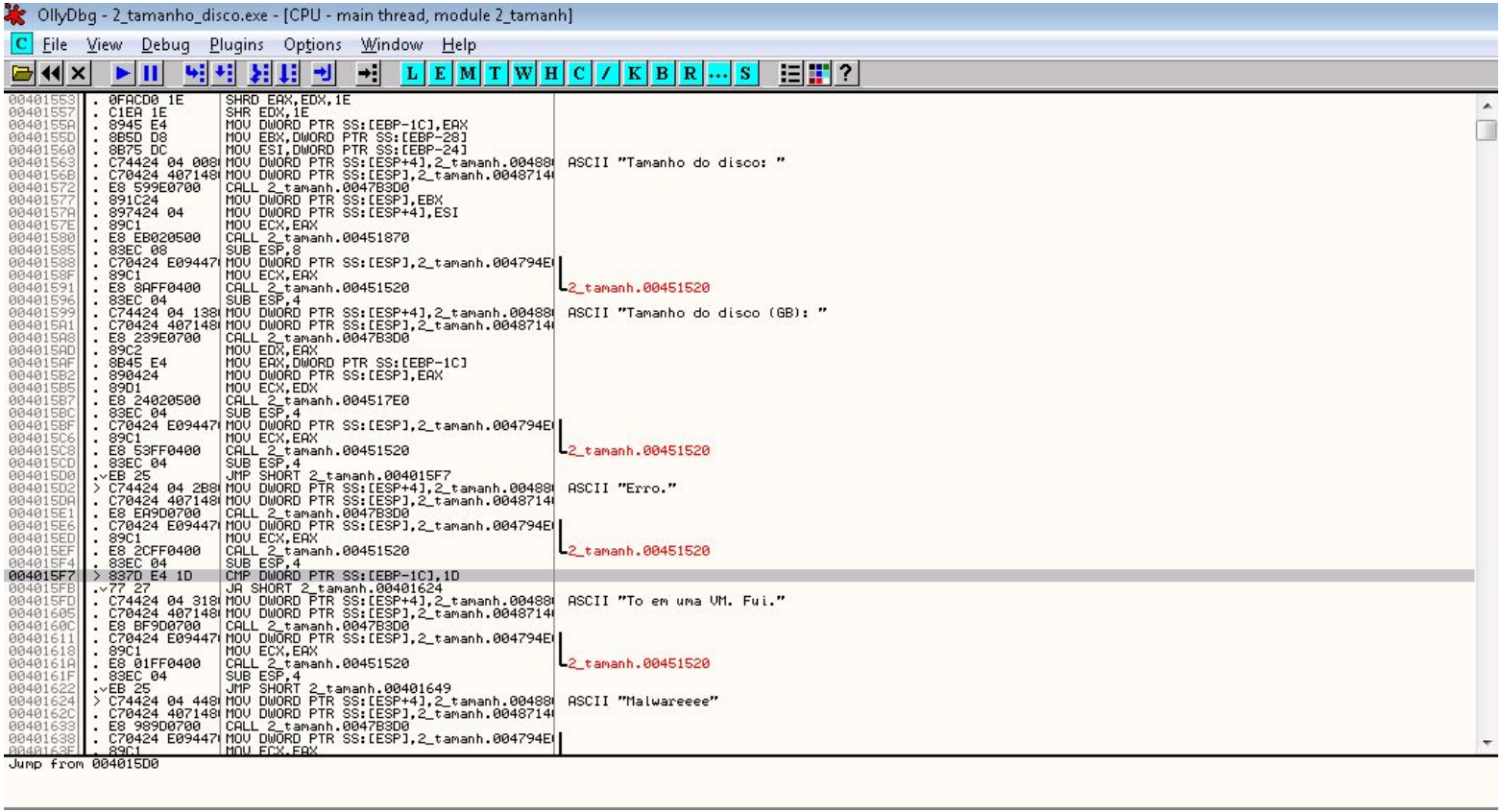
2 - Tamanho do Disco

- Disco < 30GB? VM!

```
int main() {
    ULARGE_INTEGER dskSize;
    unsigned int dskSizeGB = 0;
    if (GetDiskFreeSpaceEx(NULL, NULL, &dskSize, NULL))
        dskSizeGB = dskSize.QuadPart / (1024*1024*1024);
    if (dskSizeGB < 30)
        std::cout << "To em uma VM. Fui." << std::endl;
    else
        std::cout << "Malwareeee" << std::endl;
    return 0;
}
```

[Anti] 2 - Tamanho do Disco

2 - Tamanho do Disco



3 - Registro (Guest Additions)

- VirtualBox Guest Additions instalada?

```
int main() {
	HKEY hKey;
	LONG res = RegOpenKeyExA(HKEY_LOCAL_MACHINE,
		"SOFTWARE\\Oracle\\VirtualBox Guest Additions", 0, KEY_READ, &hKey);

	if (res == ERROR_SUCCESS)
		std::cout << "Encontrei a chave. Fui." << std::endl;
	else
		std::cout << "Rodando Malwaree" << std::endl;
}
```

[Anti] 3 - Registro (Guest Additions)

- Solução? Não instalar o Guest Additions :)

- Mas eu quero ;(
  - Então, solução da técnica 4

4 - Outras chaves no Registro

- Outras chaves no registro estão presentes
  - Mesmo sem a instalação do Guest Additions

- (HKEY_LOCAL_MACHINE, "HARDWARE\\Description\\System", **"VideoBiosVersion", "VIRTUALBOX"**)
- (HKEY_LOCAL_MACHINE, "HARDWARE\\ACPI\\DSDT**VBOX__**")
- (HKEY_LOCAL_MACHINE, "HARDWARE\\DESCRIPTION\\System", "SystemBiosDate", "**06/23/99**")
- (HKEY_LOCAL_MACHINE, "HARDWARE\\DEVICEMAP\\Scsi\\Scsi Port 0\\Scsi Bus 0\\Target Id 0\\Logical Unit Id 0", "Identifier", "**VBOX**")
- (HKEY_LOCAL_MACHINE, "SYSTEM\\ControlSet001\\Services**VBoxGuest**")
- (HKEY_LOCAL_MACHINE, "SYSTEM\\ControlSet001\\Services**VBox***")

[Anti] 4 - Outras chaves no Registro

- Solução? **Hook!**

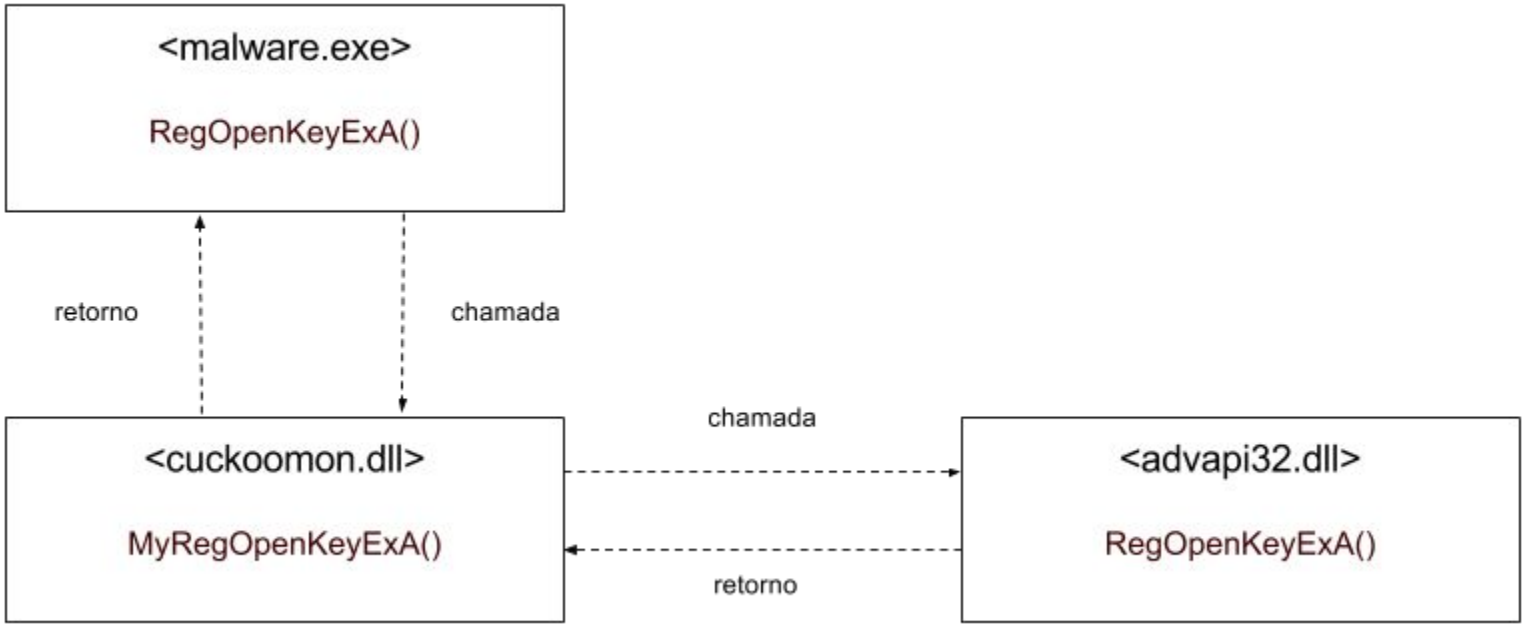
- Interceptar chamadas para API do Windows

- Modificar a resposta de acordo com os interesses [Anti-Anti-VM]
  - E logar essas chamadas

- Cuckoo injeta DLL no processo
  - cuckoomon.dll

[Anti] 4 - Outras chaves no Registro

API Hooking



5 - MAC Address

- 3 primeiros bytes do MAC Address → OUI
  - Organizationally unique identifier
  - Identifica o fabricante
  - http://standards-oui.ieee.org/oui.txt

- VirtualBox → [08:00:27]
  - Cadmus Computer Systems
- VMWare → [00:05:69, 00:0C:29, 00:1C:14, 00:50:56]
  - VMware, Inc.

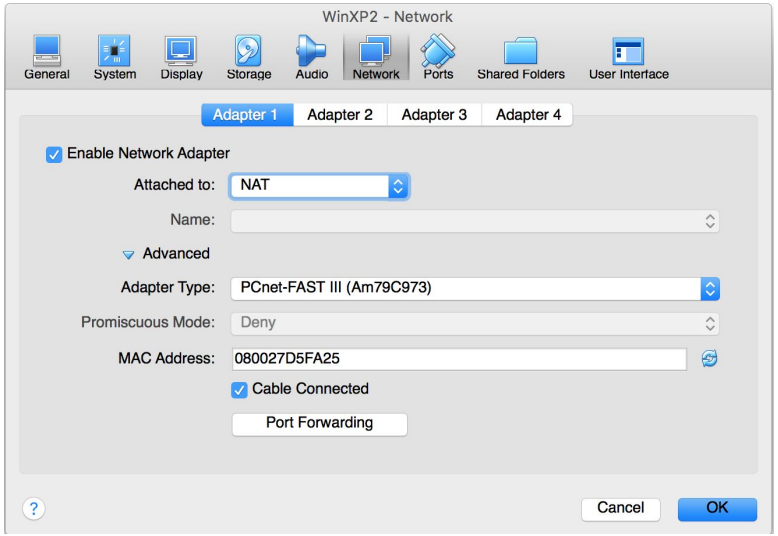
http://www.malwarestats.org/tricks.html

[Anti] 5 - MAC Address

- Troque o MAC da(s) interface(s) de rede da VM

```
$ perl -e 'for ($i=0;$i<6;$i++){@m[$i]=int(rand(256));} printf "%X:%X:%X:%X:%X:%X\n",@m;'
```

- http://www.miniwebtool.com/mac-address-generator/



6 - Mouse

- Há movimento no mouse durante um intervalo de tempo?
  - Não?? VM!!
  - Sim? #partiuroubo

```
int gensandbox_mouse_act() { // https://github.com/a0rtega/pafish/blob/master/pafish/gensandbox.c
	POINT position1, position2;
	GetCursorPos(&position1);
	Sleep(2000); /* Sleep time */
	GetCursorPos(&position2);
	if ((position1.x == position2.x) && (position1.y == position2.y)) {
		// Sem atividade durante sleep. Fui.
	}
	else {
		// Atividade durante o sleep. Vamos roubar.
	}
}
```

[Anti] 6 - Mouse

- analyzer/windows/modules/auxiliary/human.py

```
def move_mouse():
    x = random.randint(0, RESOLUTION["x"])
    y = random.randint(0, RESOLUTION["y"])

    USER32.SetCursorPos(x, y)

class Human(Auxiliary, Thread):
    def run(self):
        ....
        # only move the mouse 50% of the time, as malware can choose to act on an
        # "idle" system just as it can on an "active" system
        if random.randint(0, 3) > 1:
            click_mouse()
            move_mouse()
# https://github.com/brad-accuvant/cuckoo-modified/blob/master/analyzer/windows/modules/auxiliary/human.py
```

7 - Is Sleep() patched?

- Deixa o processo "dormindo" por um tempo antes de executar
  - Dificilmente uma VM ficará rodando por muito tempo para apenas um malware
- Sabendo disso, sandboxes normalmente fazem hook na função Sleep()
  - Para "pular" no tempo
- Podemos então verificar se o tempo está pulando

```
int gensandbox_sleep_patched() { // pafish/gensandbox.c
        DWORD time1;

        time1 = GetTickCount();
        Sleep(500);
        if ((GetTickCount() - time1) > 450 )
            vamos_roubar();
        else
            exit(0);
}
```

[Anti] 7 - Is Sleep() patched?

- Hook **GetTickCount()**!

```
HOOKDEF(DWORD, WINAPI, GetTickCount, void) {
    DWORD ret = Old_GetTickCount();

    // add the time we've skipped
    ret += (DWORD)(time_skipped.QuadPart / 10000);

    return ret;
}

// https://github.com/brad-accuvant/cuckoomon-modified/blob/MSVC/hook_sleep.c
```

Técnica 8 - Processos

- Cuckoo roda agent**.py**? E o python.exe?

- VboxSVC.exe? vboxtray.exe?

[Anti] Técnica 8 - Processos

```
protected_procname_list = [
    "vmwareuser.exe",
    "vmwareservice.exe",
    "vboxservice.exe",
    "vboxtray.exe",
    "sandboxiedcomlaunch.exe",
    "sandboxierpcss.exe",
    "procmon.exe",
    "regmon.exe",
    "filemon.exe",
    "wireshark.exe",
    "netmon.exe",
    "prl_tools_service.exe",
    "prl_tools.exe",
    "prl_cc.exe",
    "sharedintapp.exe",
    "vmtoolsd.exe",
    "vmsrvc.exe",
    "python.exe",
    "perl.exe",
]

HIDE_PIDS = set(self.pids_from_process_name_list(protected_procname_list))
```

https://github.com/brad-accuvant/cuckoo-modified/blob/master/analyzer/windows/analyzer.py

pafish

Outras técnicas

- Há muito mais!
  - Nome de usuário, nome da máquina, "shared folders", detectar GPU, …

- Há *n* formas de utilizar a mesma técnica Anti-VM
  - O mesmo serve para as Anti-Anti-VM

- Check it out: https://github.com/a0rtega/pafish

Fraquezas

- A própria evasão

- Hooking
  - Guerra constante

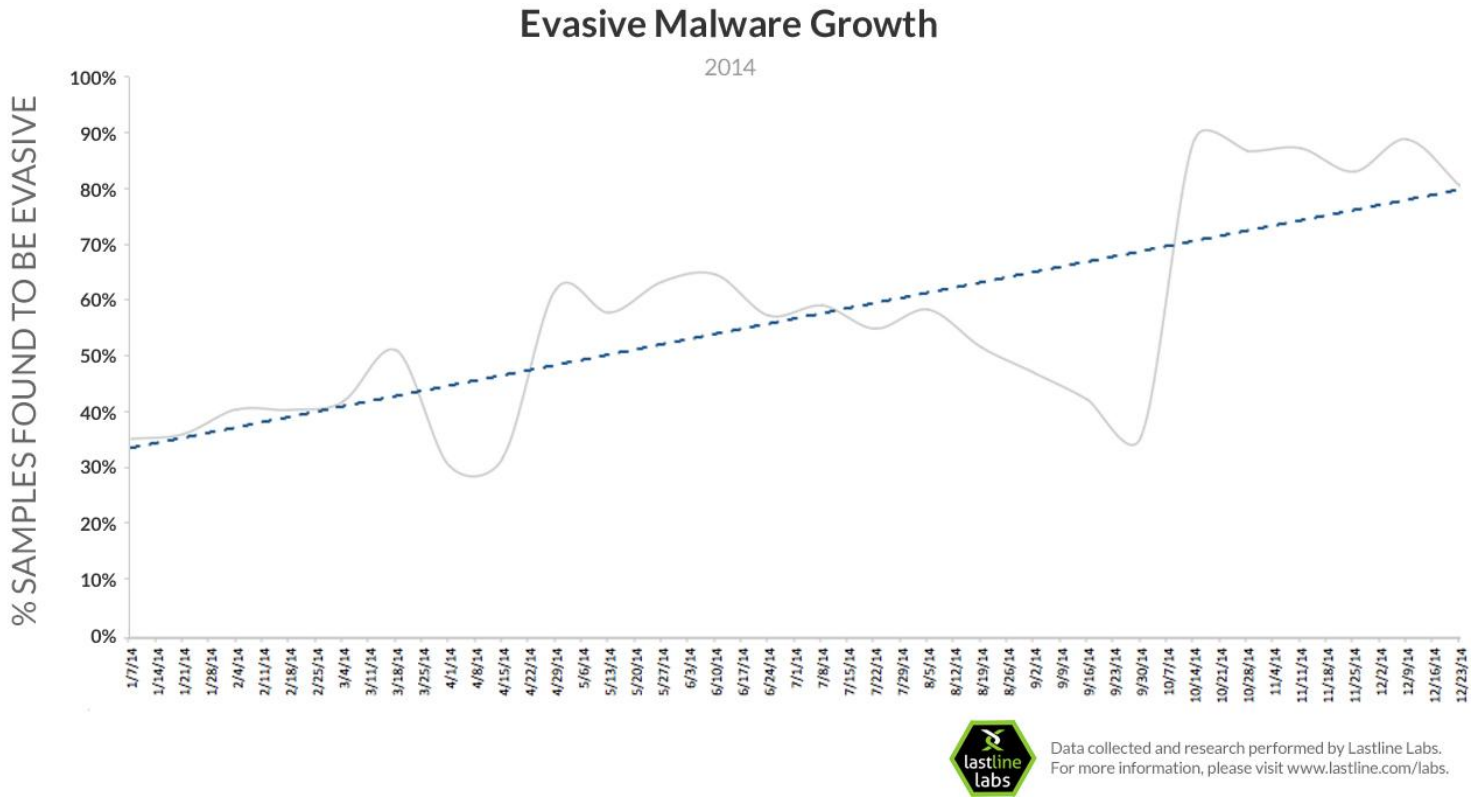
- Automação
  - Crashes?

Números

- De acordo com Chen et al., 2008. Entre 6.992 amostras:
  - 40% Reduzem suas atividades quando executadas em VM
  - 58.5% Possuem o mesmo comportamento referente a Debuggers

- https://malwr.com
  - Cuckoo :)

Números



http://labs.lastline.com/evasive-malware-gone-mainstream

Tendências

- Evasive malware são uma tendência *per si*
- Combinações de técnicas evasivas mais elaboradas
- Behavioral perceptual hash

Outras Sandboxes

- Anubis Sandbox
  - https://anubis.iseclab.org/

- Hybrid Analysis
  - https://www.hybrid-analysis.com/

Referências

- Kruegel, Christopher. "Evasive Malware Exposed and Deconstructed." RSA Conference 2015.
  - https://www.rsaconference.com/writable/presentations/file_upload/crwd-t08-evasive-malware-exposed-and-deconstructed.pdf

- Martina Lindorfer, Clemens Kolbitsch, and Paolo Milani Comparetti. 2011. Detecting environment-sensitive malware. In Proceedings of the 14th international conference on Recent Advances in Intrusion Detection (RAID'11), Robin Sommer, Davide Balzarotti, and Gregor Maier (Eds.).
  - http://www.syssec-project.eu/m/page-media/3/disarm-raid11.pdf

- Todos outros sites já citados na apresentação

https://github.com/feliperalmeida/h2hc2015-cbte

Dúvidas?

felipe.almeida@axur.com | @feliperalmeida

lucas.moura@axur.com | @mourackb