Malicious Linux Binaries: A Landscape

Lucas Galante[1],Marcus Botacin[2], André Grégio[2], Paulo Lício de Geus[1]

[1]University of Campinas (Unicamp) – {galante,paulo}@lasca.ic.unicamp.br

[2]Federal University of Paraná (UFPR) – {mfbotacin, gregio}@inf.ufpr.br

Linux Developer Conference Brazil 2019

Who Am I?

Lucas Galante

- Computer Engineering Student (EC016) @ UNICAMP
- Tracing ELF binaries since then...

Marcus Botacin

- Computer Engineer (EC010) @ UNICAMP
- Master in Computer Science (2015-2017) @ UNICAMP
- PhD Candidate (2017-???) @ UFPR

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Introduction

Motivation

- Are there Linux malware?

Reality

- Linux malware is a **real** threat!

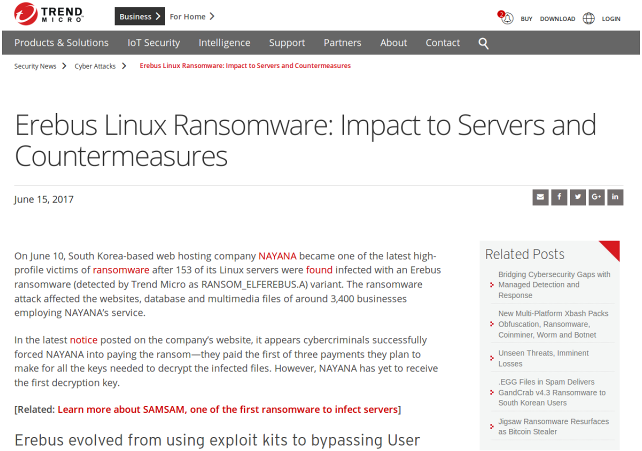
Proposal

- Understanding Linux malware samples.

Results

- Malicious Linux Binaries: A **Landscape**

Are there Linux malware?



Erebus Linux Ransomware: Impact to Servers and Countermeasures

June 15, 2017

On June 10, South Korea-based web hosting company NAYANA became one of the latest high-profile victims of ransomware after 153 of its Linux servers were found infected with an Erebus ransomware (detected by Trend Micro as RANSOM_ELFEREBUS.A) variant. The ransomware attack affected the websites, database and multimedia files of around 3,400 businesses employing NAYANA's service.

In the latest notice posted on the company's website, it appears cybercriminals successfully forced NAYANA into paying the ransom—they paid the first of three payments they plan to make for all the keys needed to decrypt the infected files. However, NAYANA has yet to receive the first decryption key.

[Related: Learn more about SAMSAM, one of the first ransomware to infect servers]

Erebus evolved from using exploit kits to bypassing User

Related Posts

- Bridging Cybersecurity Gaps with Managed Detection and Response
- New Multi-Platform Xbash Packs Obfuscation, Ransomware, Coinminer, Worm and Botnet
- Unseen Threats, Imminent Losses
- .EGG Files in Spam Delivers GandCrab v4.3 Ransomware to South Korean Users
- Jigsaw Ransomware Resurfaces as Bitcoin Stealer

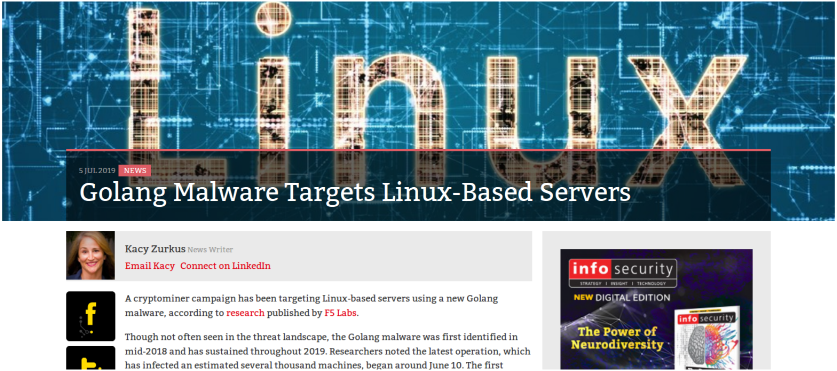
Figure: Erebus ransomware attacks South Korean internet provider.
Source: https://tinyurl.com/y5ekengt

Are there Linux malware?

HiddenWasp Malware Stings Targeted Linux Systems

Figure: Undetectable targeted remote control.
Source: https://tinyurl.com/y5mbkr2z

Are there Linux malware?



5 JUL 2019 NEWS

Golang Malware Targets Linux-Based Servers

Kacy Zurkus News Writer
Email Kacy Connect on LinkedIn

A cryptominer campaign has been targeting Linux-based servers using a new Golang malware, according to research published by F5 Labs.

Though not often seen in the threat landscape, the Golang malware was first identified in mid-2018 and has sustained throughout 2019. Researchers noted the latest operation, which has infected an estimated several thousand machines, began around June 10. The first

Figure: A cryptominer campaign written in Go!
Source: https://tinyurl.com/y2ykkmk4

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Malware Behavior Taxonomy

Table: Identified invoked system calls.

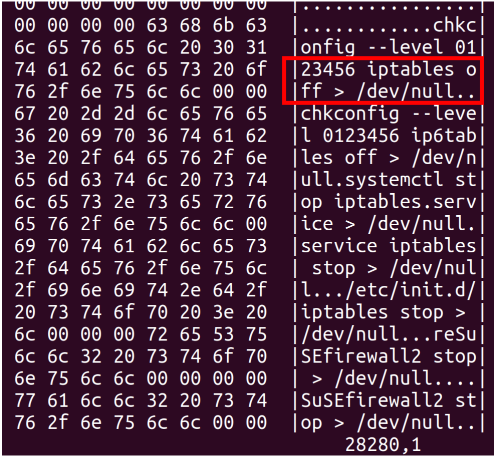
Network	Evasion	Environment	Removal	Timing	Memory	Modularity
socket	fork	gettimeofday	unlink	time	mmap	execve
connect	kill	access	rmdir	wait	munmap	fork
poll	ptrace	uname	kill	nanosleep	mprotect	clone
select		ioctl				exit
getsockname						getppid

Malware Behaviors by Examples

Listing 1: Network Scanner Malware.

```
May 13 13:21:49 lab kernel: [ 3610.320968] IN=
     OUT=ens3 SRC=192.168.122.5 DST
    =91.189.89.196
May 13 13:21:49 lab kernel: [ 3610.321356] IN=
     OUT=ens3 SRC=192.168.122.5 DST
    =91.189.89.197
May 13 13:21:49 lab kernel: [ 3610.321503] IN=
     OUT=ens3 SRC=192.168.122.5 DST
    =91.189.89.198
May 13 13:21:49 lab kernel: [ 3610.321633] IN=
     OUT=ens3 SRC=192.168.122.5 DST
    =91.189.89.199
```

Malware Behaviors by Examples



```
00 00 00 00 00 00 00 00  |................|
00 00 00 00 63 68 6b 63  |............chkc|
6c 65 76 65 6c 20 30 31  |onfig --level 01|
74 61 62 6c 65 73 20 6f  |23456 iptables o|
76 2f 6e 75 6c 6c 00 00  |ff > /dev/null..|
67 20 2d 2d 6c 65 76 65  |chkconfig --leve|
36 20 69 70 36 74 61 62  |l 0123456 ip6tab|
3e 20 2f 64 65 76 2f 6e  |les off > /dev/n|
65 6d 63 74 6c 20 73 74  |ull.systemctl st|
6c 65 73 2e 73 65 72 76  |op iptables.serv|
65 76 2f 6e 75 6c 6c 00  |ice > /dev/null.|
69 70 74 61 62 6c 65 73  |service iptables|
2f 64 65 76 2f 6e 75 6c  | stop > /dev/nul|
2f 69 6e 69 74 2e 64 2f  |l.../etc/init.d/|
20 73 74 6f 70 20 3e 20  |iptables stop > |
6c 00 00 00 72 65 53 75  |/dev/null...reSu|
6c 6c 32 20 73 74 6f 70  |SEfirewall2 stop|
6e 75 6c 6c 00 00 00 00  | > /dev/null....|
77 61 6c 6c 32 20 73 74  |SuSEfirewall2 st|
76 2f 6e 75 6c 6c 00 00  |op > /dev/null..|
                                28280,1
```

Figure: Network Exfiltrator Malware.

Malware Behaviors by Examples

Listing 2: Process Terminator Malware.

```
[pid 11048] execve(”/bin/sh”, [”sh”, ”−c”, ”
    killall b−server”]
[pid 11049] execve(”/usr/bin/killall”, [”
    killall”, ”b−server”]
[pid 11051] kill(11046, SIG_0)              = 0
[pid 11051] kill(11046, SIG_0)              = 0
[pid 11046] kill(11051, SIG_0)              = 0
```

Malware Behaviors by Examples

Listing 3: Modular Malware.

```
execve("./malware.bin", ["./malware.bin"]
execve("/bin/sh", ["./malware.bin", "-c", "
    exec './malware.bin' \"$@\"", "./malware.
    bin"]
execve("/bin/sh", ["./malware.bin", "-e", "-c
    ", "#!/bin/sh -e\nclear\n\nbash=$(echo
    "..., "./malware.bin"]
[pid 11045] execve("/usr/bin/clear", ["clear"]
```

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

(Anti-)Analysis Techniques

Evasion Countermeasures

Table: Adopted strategy to handle evasive samples.

Technique	Tool	Evasion	Countermeasure
Static analysis	*objdump* *file* *strings*	obfuscation	Dynamic analysis
Dynamic analysis	*ltrace*	Static compilation	*ptrace* step-by-step
	ptrace	*ptrace* check	binary patching
	strace	Long *sleep*	*LD_PRELOAD*
	LD_PRELOAD	Injection blocking	Kernel *hooks*

Hands On Examples

Obfuscation

- `upx -1 <binary>`

Hidden Artifacts

- `ltrace <gcc -static <binary>>`

Anti-Debug

- `if(ptrace(PTRACE_TRACEME)==-1)`

Analysis Delays

- `sleep(LOOOOOONG_TIME)`

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Rootkit Examples

- **ls**: Hidding a string.
- **ps**: Hidding a string.
- **stat**: Hidding an inode.

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Binaries Architectures

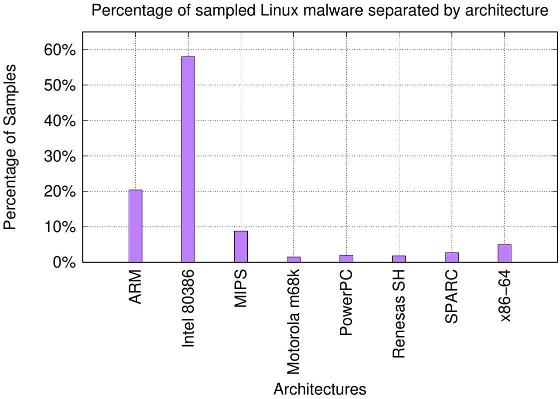


Figure: ELF binary samples distributed by architectures.

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Objdump

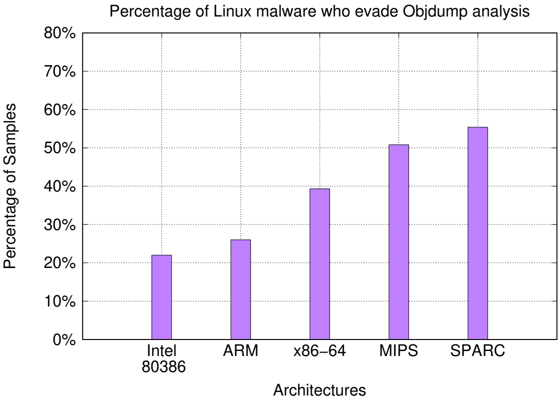


Figure: Percentage of malware that failed to dissasembly.

Static Functions

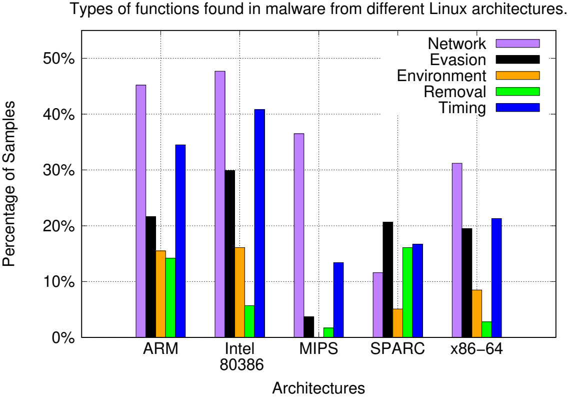


Figure: Malware behavior prevalence by malware architectures.

Network Strings

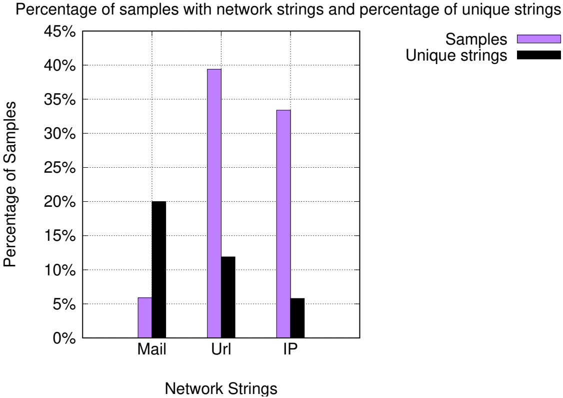


Figure: Network-Related Strings. Rate of samples with network related strings.

Packer

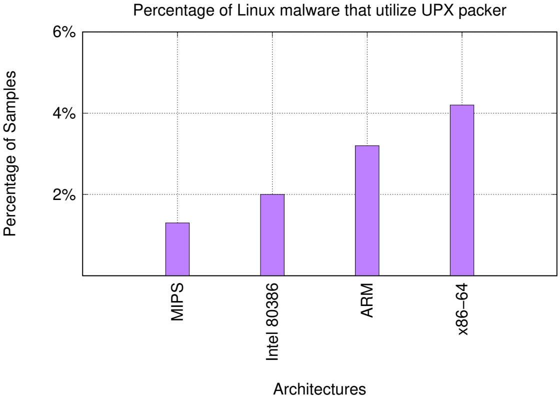


Figure: Rate of UPX-packed samples. Few samples are packed.

AV Labels

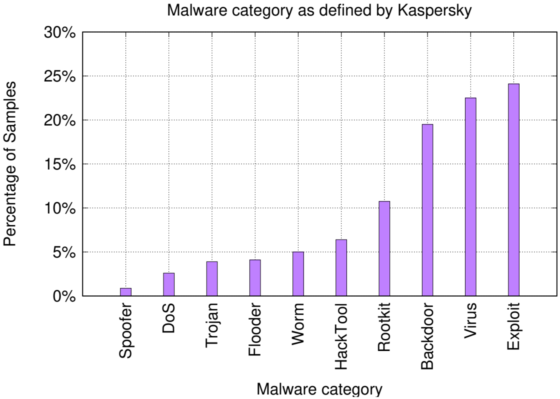


Figure: AV labels according Kaspersky AV. We observe a prevalence of exploits

Clusters

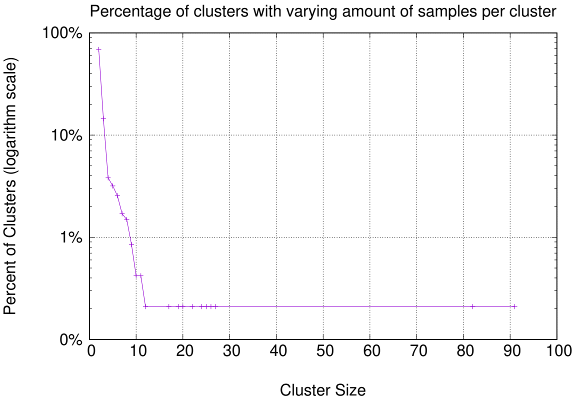


Figure: Samples variants clustering. Smaller clusters are prevalent.

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Timeout Signals

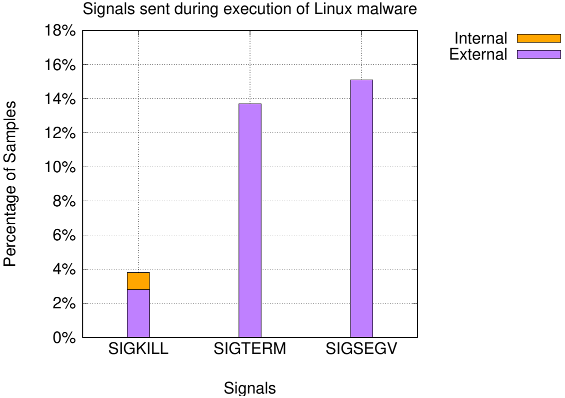


Figure: Observed Signals during execution.

Behavior

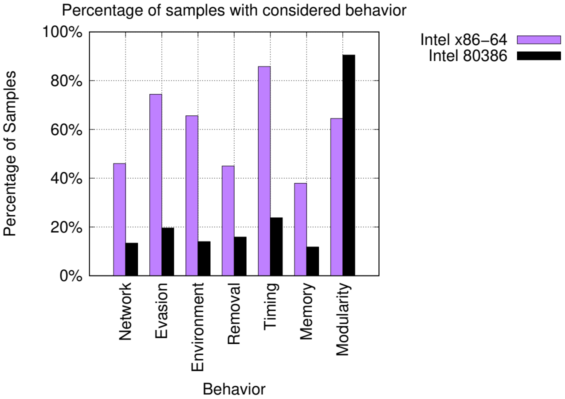


Figure: Malware behavior prevalence.

Acessed Files

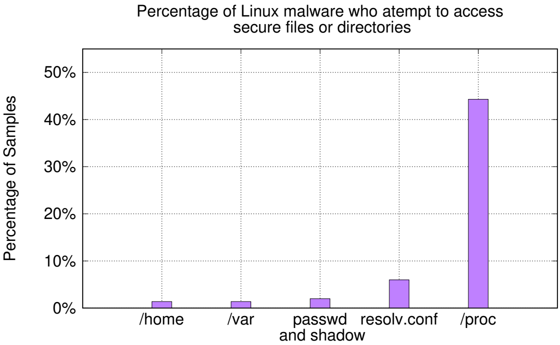


Figure: Accessed files and directories.

I/O Operations

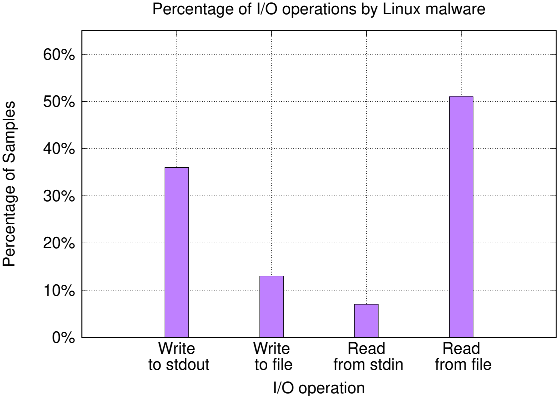


Figure: I/O operations. Most samples do not present direct user interaction.

Evasion

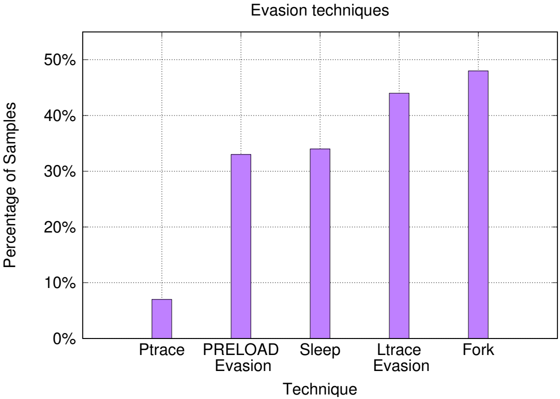


Figure: Evasion Techniques. Samples present diversified evasion methods.

Network

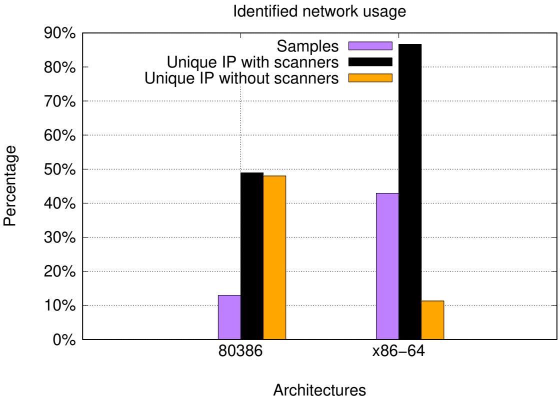


Figure: Identified network usage. Scanners dominate unique IP rate.

Domains

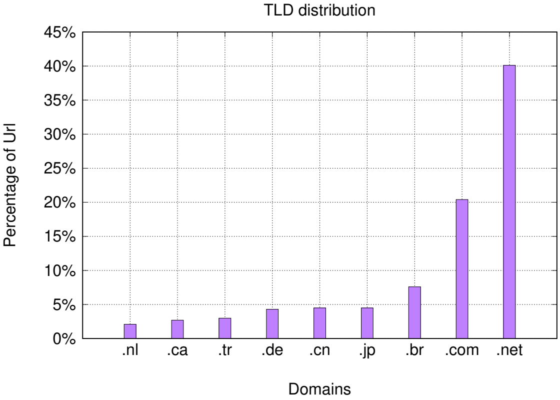


Figure: TLD distribution. Global domains are prevalent. Local domains are present due to scanners enumeration.

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Malware Classification

Table: Accuracy rates for Random Forest classifier.

Max Depth/ Estimators (#)	16	32	64
8	99.26%	99.26%	99.26%
16	99.15%	**99.36%**	99.28%
32	99.26%	99.26%	99.31%

Feature Importance

Table: Feature importance on malware behavior classification.

Static			
Discrete		Continuous	
Network strings	40%	Binary size	27%
UPX present	17%	# headers	16.70%
passwd strings	1.40%	# debug sections	0.20%

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Linux AV Labels

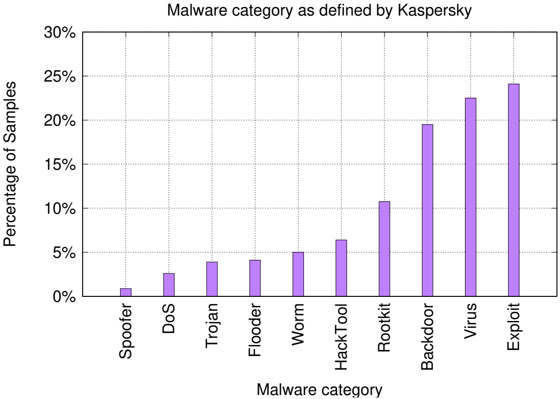


Figure: AV labels according Kaspersky AV. We observe a prevalence of exploits

Windows AV Labels

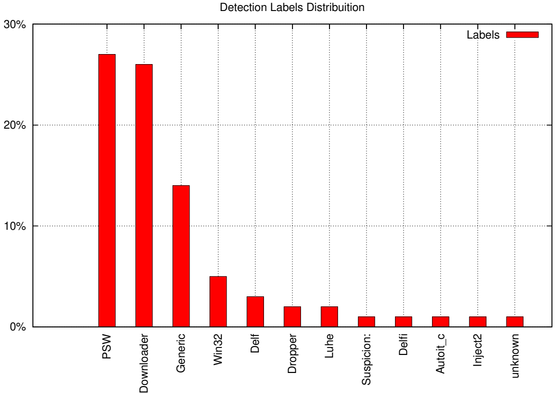


Figure: AV labels for Windows malware.

Linux Evasion Techniques

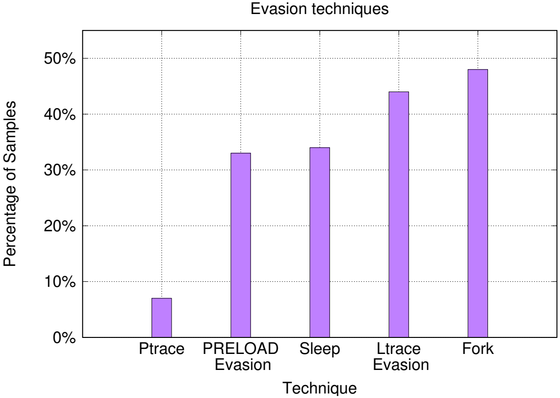


Figure: Evasion Techniques. Samples present diversified evasion methods.

Windows Evasion Techniques

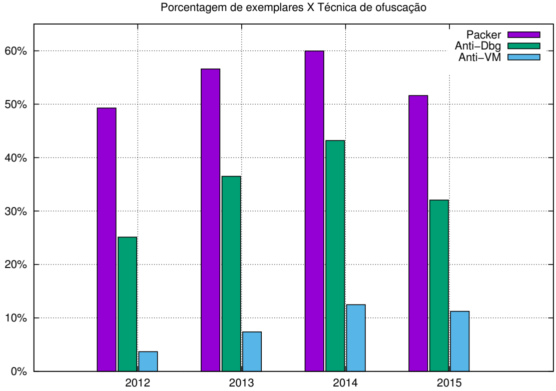


Figure: Windows malware evasion techniques over time.

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

SSH Backdoor

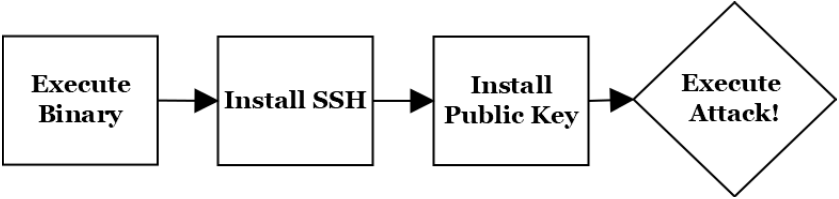


Figure: Execution flow of backdoor malware with SSH injection.

SSH Backdoor

Listing 4: Backdoor sample in action. It drops attacker key into the system, thus granting remote access.

```
malloc(381) = 0x2083c60
strlen("PPK\016QPB\003bbbba\020mYB`\022Z@\021
    fbbbbgbrba"...)
strcat("", "ssh-rsa AAAAB3NzaC1yc2EAAAADAQAB"...)
```

Erebus

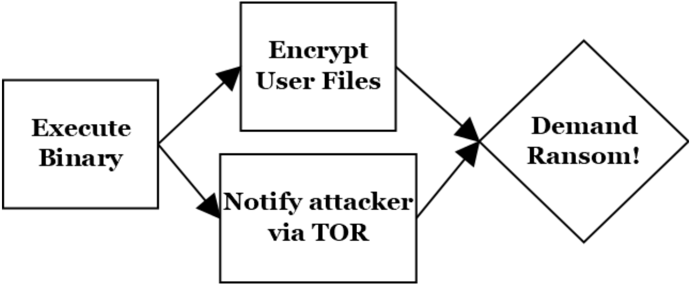


Figure: Execution flow of Erebus ransomware.

Erebus

Listing 5: Erebus Execution. It connects to runtime-generated IP addresses and to TOR-based hidden services and onion domains.

```
strncmp(""—————BEGIN PUBLIC KEY—————\\nMII"..., "
    null", 4)
strncmp("3,"tg":"216.126.224.128\\/24","bu"..., "
    null", 4)
strncmp(""7fv4vg4n26cxleel.hiddenservice."..., "
    null", 4)
strncmp(""qzjordhlw5mqhcn7.onion.to","qzj"..., "
    true", 4)
```

Agenda

1. Introduction
   - Introduction
2. Behaviors & Implementations
   - Methodology
3. Landscape
   - Dataset
   - Static Analysis
   - Dynamic Analysis
   - Comparion Scenarios
   - Case Studies
4. Conclusions
   - Conclusions

Conclusions

Lessons Learned

- The threat of Linux malware is real.
- Linux malware are able to infect multiple systems.
- They present an intense use of network resource.
- They rely on diverse analysis evasion techniques.

Questions & Comments?

Contact

- **galante@lasca.ic.unicamp.br**
- **mfbotacin@inf.ufpr.br**

Academic Paper

- L. Galante, M. Botacin, A. Grégio, P. Geus, *Malicious Linux Binaries: A Landscape*, SBSeg 2018

Additional Material

- `https://github.com/marcusbotacin/Linux.Malware`