MALWARE SIGNATUREN

Dipl.-Inform. Julian Dammann

Dipl.-Inform. Peter Weidenbach

julian.dammann@fkie.fraunhofer.de,
peter.weidenbach@fkie.fraunhofer.de

Fraunhofer
FKIE

MALWARE SIGNATUREN

1. *Signaturen Allgemein*
2. *Einführung ClamAV*
3. *ClamAV-Signaturen im Detail*
   a. *Hash-basierte Signaturen*
   b. *Byte-Pattern-basierte Signaturen*
   c. *Container-basierte Signaturen*
4. *ClamAV Besonderheiten*
5. *Fun-Facts*
6. *Zusammenfassung*

1. Signaturen allgemein – Was ist eine Signatur?

- klassische Definition (static signature):

  "Sequences of bytes, which (hopefully) uniquely characterize the virus."[1]

- statische Heuristiken (static heuristic):

  "Static heuristics can find known or unknown viruses
  by looking for pieces of code that are generally 'virus-like,'
  instead of scaning for specific virus signatures."[1]

- Verhaltensmonitor/-blocker (dynamic signature/heuristic):

  "A behavior blocker watches for a program to stray
  from what the blocker considers to be 'normal' behavior."[1]

[1] Aycock, John; Computer Viruses and Malware; Springer 2006

2. Einführung ClamAV

- OpenSource Virenscanner
- Hauptentwickler: SourceFire (Cisco)
- Einordnung: Statische Signatur / statische Heuristik
- Plattformen: Linux, BSD, Windows, …
- Erstes Release: 2002
- LibClamAV → Integration in eigene Software möglich
- “It is the de facto standard for mail gateway scanning.” [2]

[2] www.clamav.net

3. ClamAV-Signaturen im Detail

- basiert auf mehreren Signaturformaten
  - Hash-basierte Signaturen (.hdb, .mdb)
  - Byte-Pattern-basierte Signaturen (.nbd, .lbd)
  - Countainer-basierte Signaturen (.cdb)
- es gibt eine Hash-basierte Whitelist (.fp)

3.a Hash-basierte Signaturen – Ganze Dateien

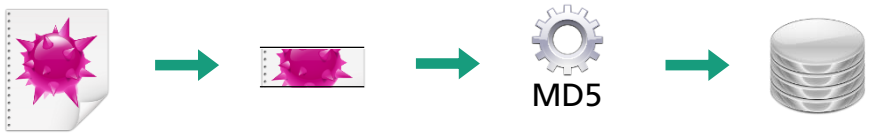

```
$ sigtool --md5 hello_world.exe >> virusdatenbank.hdb
```

48c4533230e1ae1c118c741c0db19dfb:17387:hello_world.exe

MD5-Hash | Größe | Name

- Dateiendung: .hdb

3.a Hash-basierte Signaturen – Section




```
$ sigtool --mdb part_of_exe >> virusdatenbank.mdb
```

```
17387:48c4533230e1ae1c118c741c0db19dfb:part_of_exe
```

PE-Section Größe

MD5-Hash

Name

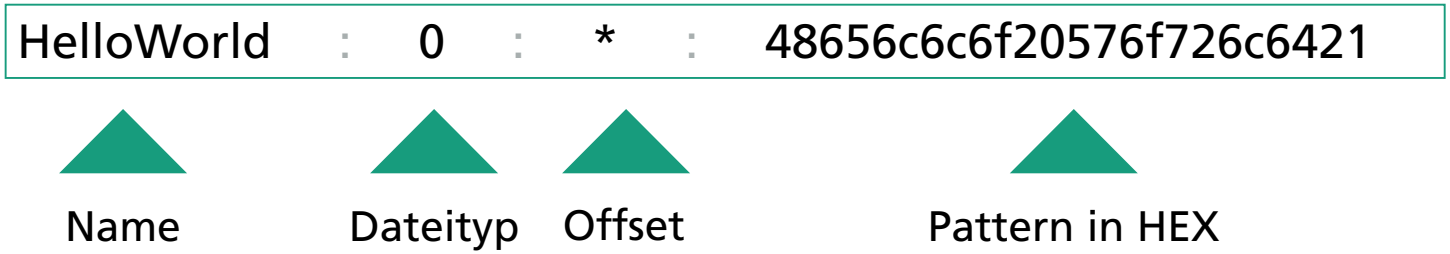
- Dateiendung: .mdb

3.a Hash-basierte Signaturen

- Nachteile:
  - bedingt geeignet für polymorphe Malware
  - Whitelist teilweise nutzlos

3.b Byte-Pattern-basierte Signaturen – Extended Signature Format

- Idee: Durchsuche die Datei byteweise nach Pattern



- Dateiendung: .ndb

3.b Byte-Pattern-basierte Signaturen – Extended Signature Format

```
HelloWorld : 0 : * : 48656c6c6f20576f726c6421
```

- 0 → beliebig
- 1 → Windows PE
- 2 → Datei innerhalb eines OLE Containers
- 3 → HTML
- 4 → Email
- 5 → Bild
- 6 → ELF
- 7 → ASCII Text
- 8 → Reserviert
- 9 → Mach-O

3.b Byte-Pattern-basierte Signaturen – Extended Signature Format

```
HelloWorld : 0 : * : 48656c6c6f20576f726c6421
```

n := [0-9]*

*	→	beliebig
n	→	absoluter Offset in Bytes
EOF-n	→	Offset vom Ende der Datei gerechnet

Nur für Signaturen von PE, ELF und Mach-O Dateien:

EP+n	→	Entry Point + n Bytes
EP-n	→	Entry Point – n Bytes
Sx+n	→	Start von Section Xs Daten + n Bytes
Sx-n	→	Start von Section Xs Daten – n Bytes
SL+n	→	Start von letzter Section + n Bytes
SL-n	→	Start von letzter Section – n Bytes

3.b Byte-Pattern-basierte Signaturen – Extended Signature Format

```
HelloWorld : 0 : * : 48656c6c6f20576f726c6421
```

Floating Offset:

- Syntax: Offset, MaxShift
- Bedeutung: Startposition des Patterns liegt zwischen Offset und Offset + MaxShift
- Beispiel: EP+5,10

3.b Byte-Pattern-basierte Signaturen – Extended Signature Format

```
HelloWorld : 0 : * : 48656c6c6f20576f726c6421
```

Wildcards:

??	→	beliebiges Byte
?a	→	beliebige vier low bits
a?	→	beliebige vier high bits
*	→	beliebige Anzahl an beliebigen Bytes
{n}	→	n beliebige Bytes
{-n}	→	n oder weniger beliebige Bytes
{n-}	→	n oder mehr beliebige Bytes
{n-m}	→	zwischen n und m beliebige Bytes
(aa|bb)	→	aa oder bb
!(aa|bb)	→	beliebiges Bytes außer aa und bb

3.b Byte-Pattern-basierte Signaturen – Logical Signatures

- Idee: Verbinde verschiedene Bytepattern mit logischen Verknüpfungen

```
HelloWorld;Target:0;(0&1)|2;*:48656c6c6f;*:20576f726c64;*:21
```

Name | Ziel | Logischer Ausdruck | Offset:Pattern in HEX

- Dateiendung: .ldb

3.b Byte-Pattern-basierte Signaturen – Logical Signatures

```
HelloWorld;Target:0;(0&1)|2;*:48656c6c6f;*:20576f726c64;*:21
```

Name | Ziel | Logischer Ausdruck | Offset:Pattern in HEX

Segmente werden mit Komma getrennt:

- Target: → Dateityp (siehe ESF)
- Engine: → benötigte Engine Funktionalität
- FileSize: → Dateigröße in Bytes
- EntryPoint: → Entry Point Offset in Bytes
- NumberOfSections: → Anzahl der Sections
- Container: → Dateityp des Containers

3.b Byte-Pattern-basierte Signaturen – Logical Signatures

```
HelloWorld;Target:0;(0&1)|2;*:48656c6c6f;*:20576f726c64;*:21
```

Name | Ziel | Logischer Ausdruck | Offset:Pattern in HEX

A & B	→	A und B kommen vor
A | B	→	A oder B kommt vor
A = X	→	A kommt X mal vor
A > X	→	A kommt mehr als X mal vor
A < X	→	A kommt weniger als X mal vor
(A|B|…) > X,Y	→	(A|B|…) kommt mehr als X mal vor und mindestens Y verschiedene aus A, B, … kommen vor

3.b Byte-Pattern-basierte Signaturen – Logical Signatures

```
HelloWorld;Target:0;(0&1)|2;*:48656c6c6f;*:20576f726c64;*:21
```

Name

Ziel

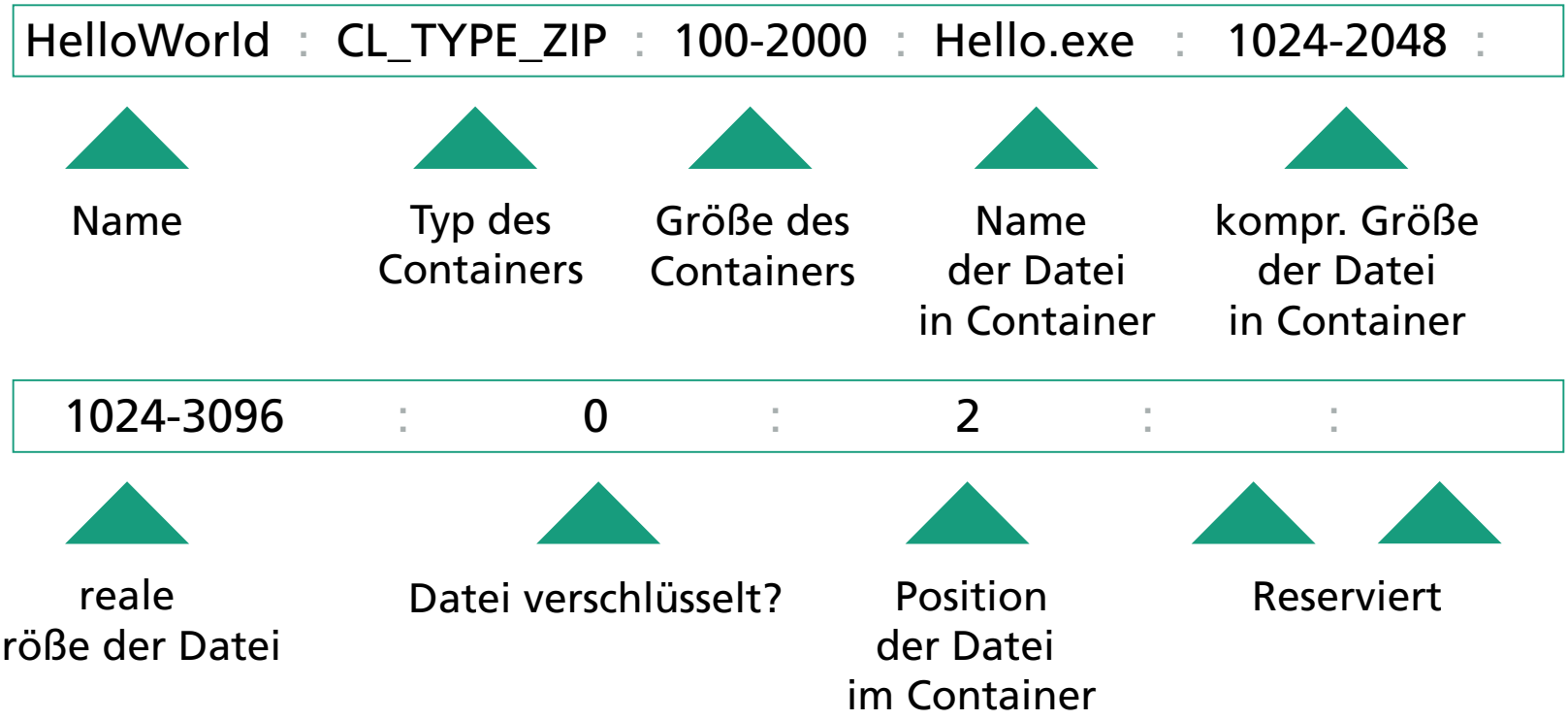
Logischer Ausdruck

Offset:Pattern in HEX

siehe ESF

3.c Container-basierte Signaturen

- Idee: Betrachte Metadaten von Datei-Containern



- Dateiendung: .cdb

4 ClamAV Besonderheiten - Normierung

- Idee: HTML- und Text-Dateien werden normiert
- mehrfache Leerzeichen werden entfernt
- alle Buchstaben werden klein geschrieben
- zusätzlich bei HTML-Dateien:
  - Kommentare werden entfernt
  - JScript.encode Teile werden dekodiert
  - Zeichenrefernzen (Bsp: f) werden dekodiert
  - wird mit und ohne HTML-Tags betrachtet

4 ClamAV Besonderheiten – Komprimierte Dateien

- Idee: Dekomprimiere vor der Analyse
- analysiert Dateien in Containern
- entpackt gepackte PE-Dateien:
  - unterstütze Packer:

    Aspack, UPX, FSG, Petite, PeSpin, NsPack,

    wwpack32, Mew, Upack, Yoda Cryptor

4 ClamAV Besonderheiten - Whitelist

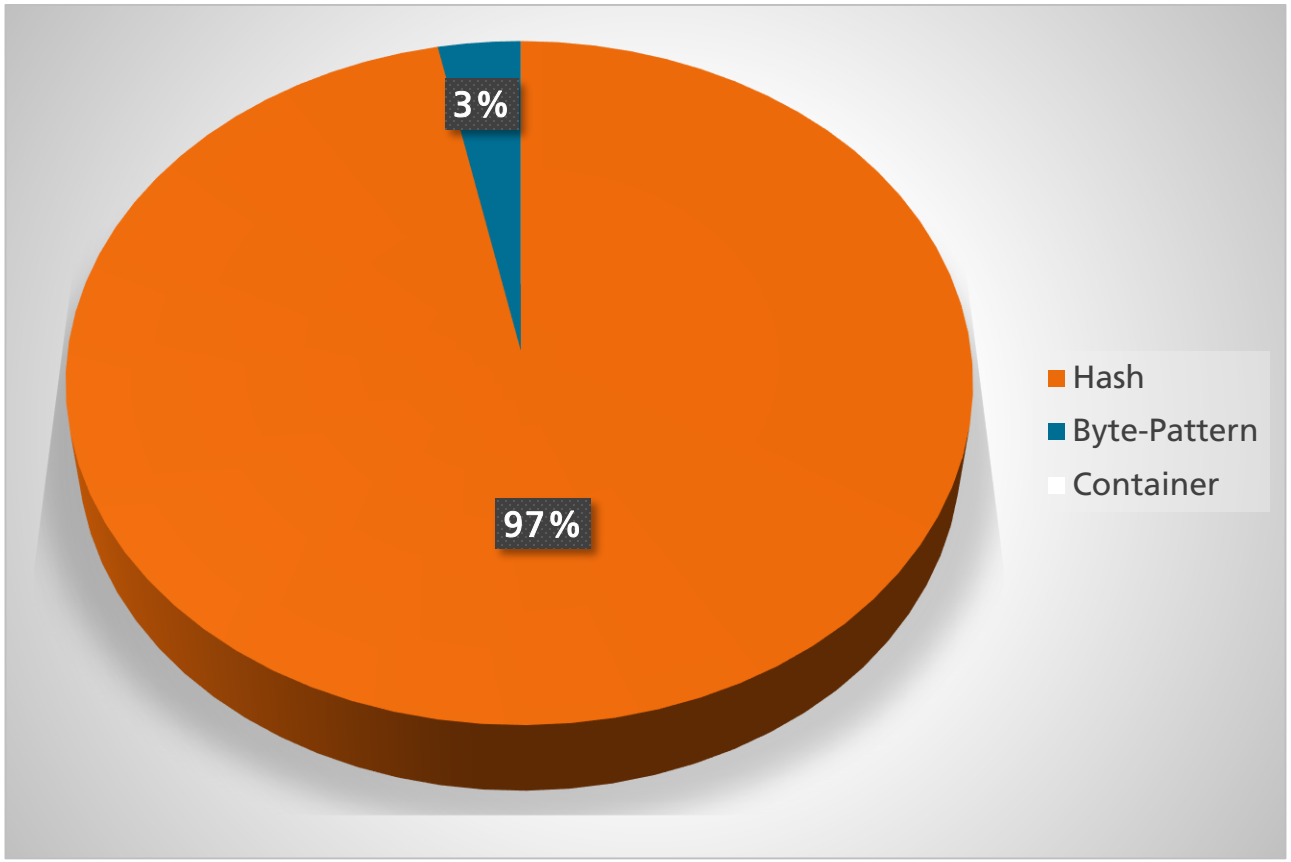
- Idee: False Positives durch Whitelist verhindern
- Hash und Größe von „guter“ Datei werden genutzt
- Format wie bei Hash-basierten Signaturen für ganze Dateien
- Dateiendung: .fp

- „ClamAV ist berüchtigt dafür, dass es häufig auch auf harmlose Programme anspringt.“ [3]

[3] http://www.heise.de/ct/hotline/FAQ-Desinfec-t-2013-2056614.html

5 Fun-Facts

- Signaturen in der ClamAV Datenbank: 3.101.896 (≈ 200MB)



(Stand: 03.02.14)

5 Fun-Facts

- akzeptable FP-Rate bei Symantec < 0,1%

6 Zusammenfassung

- Signaturen für verschiedene Aufgaben
- Beispiel Virenscanner: ClamAV
  - Verschiedene Signaturformate
    - Hash-basiert
    - Byte-Pattern-basiert
    - Container-basiert
  - Präanalyse:
    - Normierung von Text- und HTML-Dateien
    - Entpacken von Dateien in Containern
    - Entpacken von PEs
  - Whitelist zur False-Positive-Minimierung
- Hash-basierte Signaturen sind heute wenig sinnvoll