Verwaltung von Signaturen für Malware-Gruppen

SPRING 2010

5. Graduierten-Workshop über Reaktive Sicherheit

Sebastian Uellenbeck, Michael Meier

Informationssysteme und Sicherheit (ISSI)
Lehrstuhl VI
Fakultät für Informatik
TU Dortmund

07. Juli 2010

Gliederung

Gliederung

Malware und Signaturen

Malware = Malicious Software

Oft ähnliches Verhalten, aber unterschiedliche Binaries durch:

- Oligomorphismus (# Dekoder begrenzt)
- Polymorphismus (# Dekoder unbegrenzt)
- Metamorphismus (äquivalente Systemrufe)

Malware-Gruppe ist Menge von Malware mit ähnlichem Verhalten.

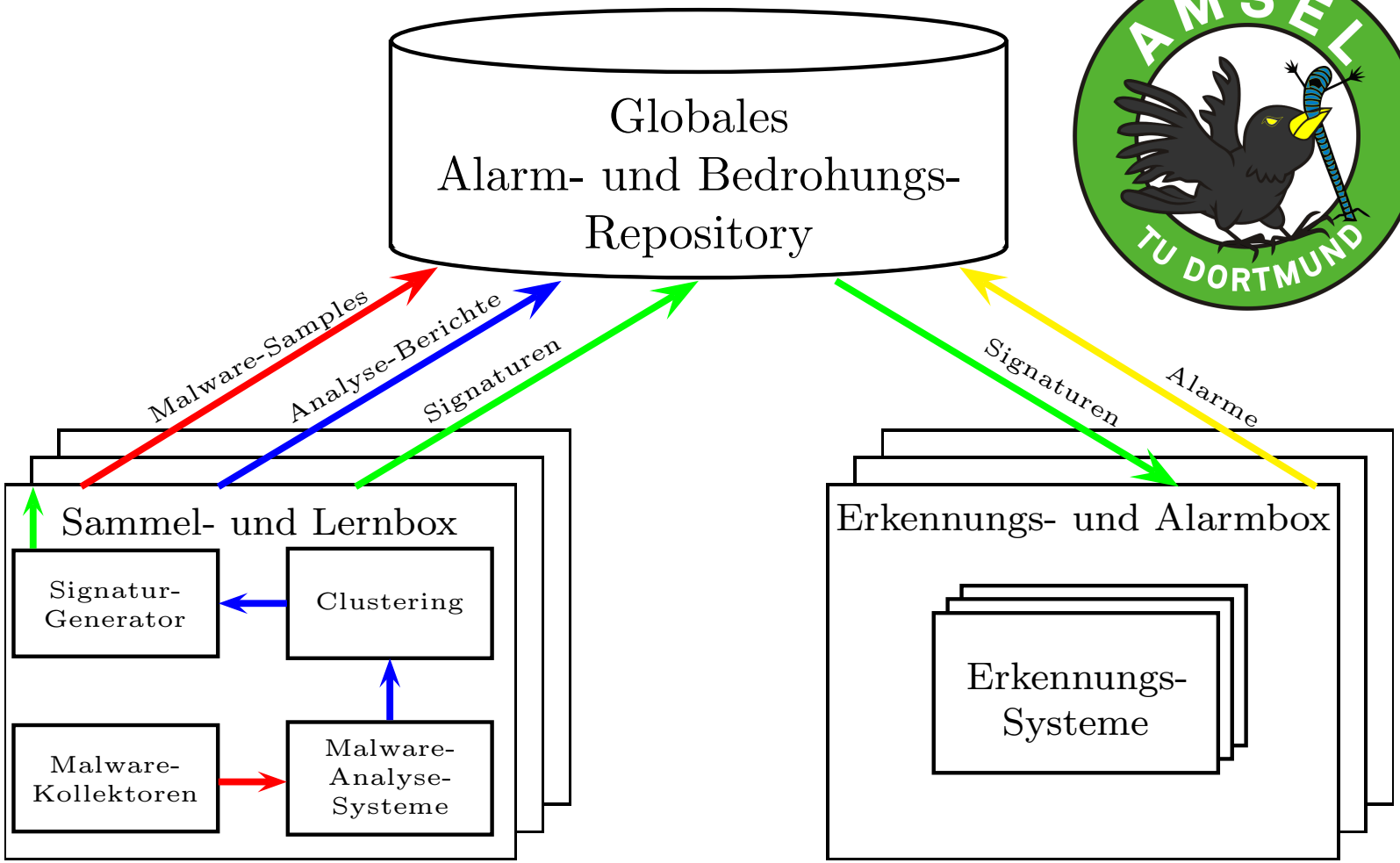
Signaturen

Sammlung eindeutiger Merkmale zur Erkennung von Malware-Gruppen:

- Musterbasiert: Dateigröße, enthaltene Bytefolgen
- Verhaltensbasiert: Systemrufe, Netzwerkverkehr

AMSEL-System und Kommunikationswege

AMSEL = **A**utomatisch **M**alware **S**ammeln und **E**rkennen **L**ernen



Begriffe

Goodpool	Menge von gutartigem Verhalten.
Signaturbasis	Menge von unterschiedlichen Signaturen, die sich zu einem bestimmten Zeitpunkt in einer Komponente (SL-Box, ...) befindet.
Verhalten	Das Verhalten einer Signatur ist die Menge der durch die Signatur abgedeckten Verhaltensberichte.

Konventionen

Durch Signatur abgedecktes Verhalten

- ist durch Verhaltensberichte definiert.
- bezieht sich auf den Goodpool, gegen den Signatur geprüft wurde.

Wichtig

Nur Signaturen, die gegen den gleichen Goodpool geprüft wurden, können vergleichend untersucht werden!

Motivation

Warum Signaturverwaltungsmechanismen untersuchen?

- Signaturbasis ist dynamisch
- konkrete Signatur kann
  - nachträglich verändert werden (neue Version)
  - sich nach der Verteilung als ungeeignet herausstellen
- globales Bedrohungs-Repository soll Signaturen aus mehreren SL-Boxen effizient an EA-Boxen verteilen
- Anzahl SL- und EA-Boxen kann beliebig groß werden

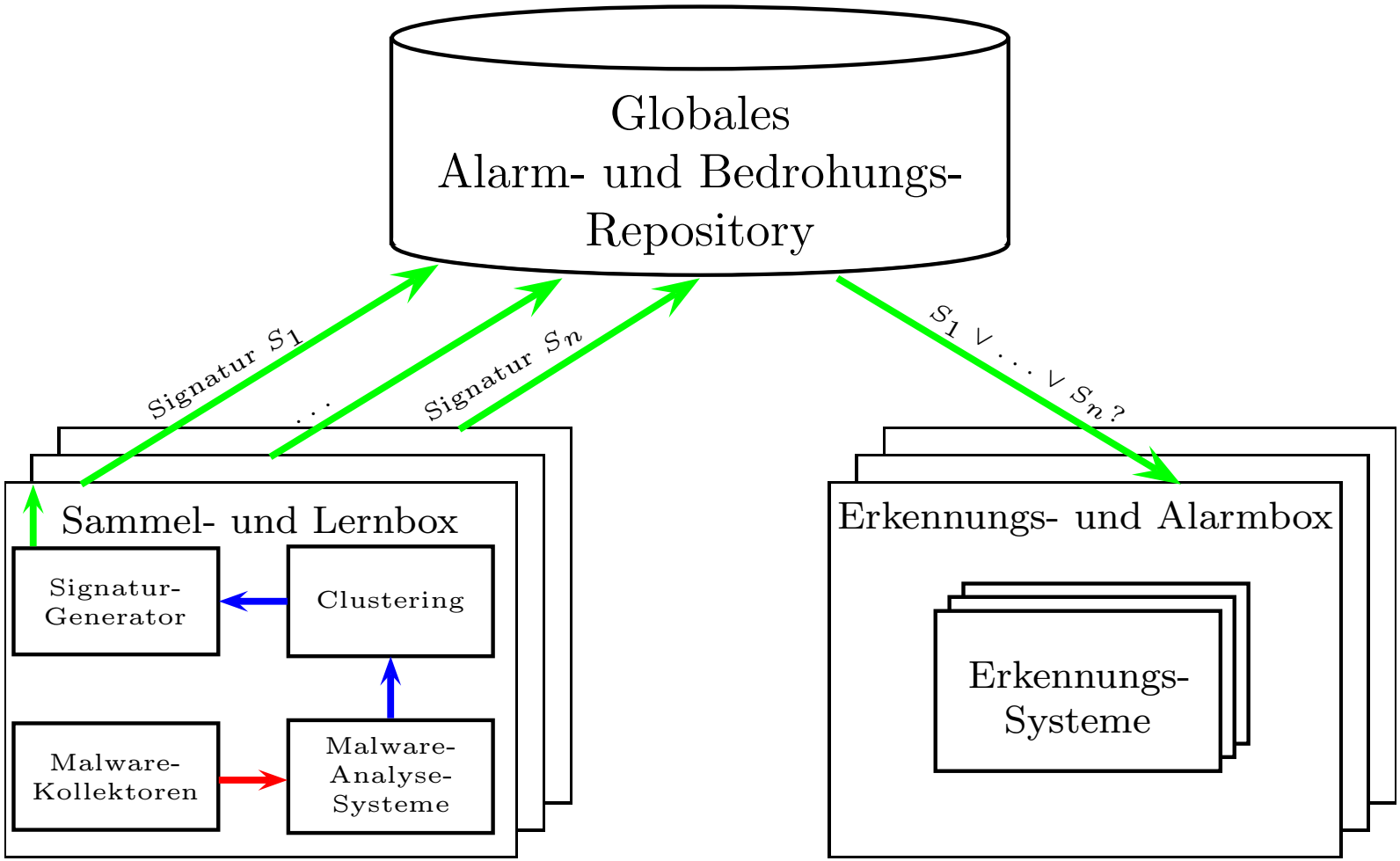
Gliederung

Verwaltungsaufgaben im Überblick

1. Verteilung
   - Aktualisierung (vorhandener Signaturbasen)
   - Rückruf (ungeeigneter Signaturen)
2. Versionierung
3. Revisionierung
4. Signaturanalyse

Verteilung: Aktualisierung und Rückruf

Welche Informationen werden zwischen den Komponenten übertragen?



Versionierung und Revisionierung

Versionierung

Wie können Signaturen global eindeutig identifiziert werden?

→ global eindeutige Identifikatoren

→ globale Versionsnummern

- Vergabe durch zentrale Instanz (globales Bedrohungs-Repository)

Revisionierung

Wie können Signaturen lokal eindeutig in SL-Box identifiziert werden?

- Nutzung von lokalen Revisionsnummern
- Vergabe durch einzelne Instanzen von SL-Boxen

Signaturanalyse 1/2

Ziel: Differentielle Aktualisierung einer Signaturbasis

Existieren Abhängigkeiten/Beziehungen zwischen Signaturen $\tilde{S}_i$ und $\tilde{S}_j$ aus verschiedenen SL-Boxen?

- Zwischen abgedeckten Verhaltensmengen $V(\tilde{S}_i)$ und $V(\tilde{S}_j)$ können Mengenbeziehung bestimmt werden.
- Ist abgedeckte Verhaltensmenge von $\tilde{S}_i$ eine Obermenge von $\tilde{S}_j$, so wird $\tilde{S}_j$ nicht benötigt:

$$V(\tilde{S}_j) \subseteq V(\tilde{S}_i)$$

$\rightarrow$ Reduktion der Anzahl der zu verteilenden Signaturen möglich.

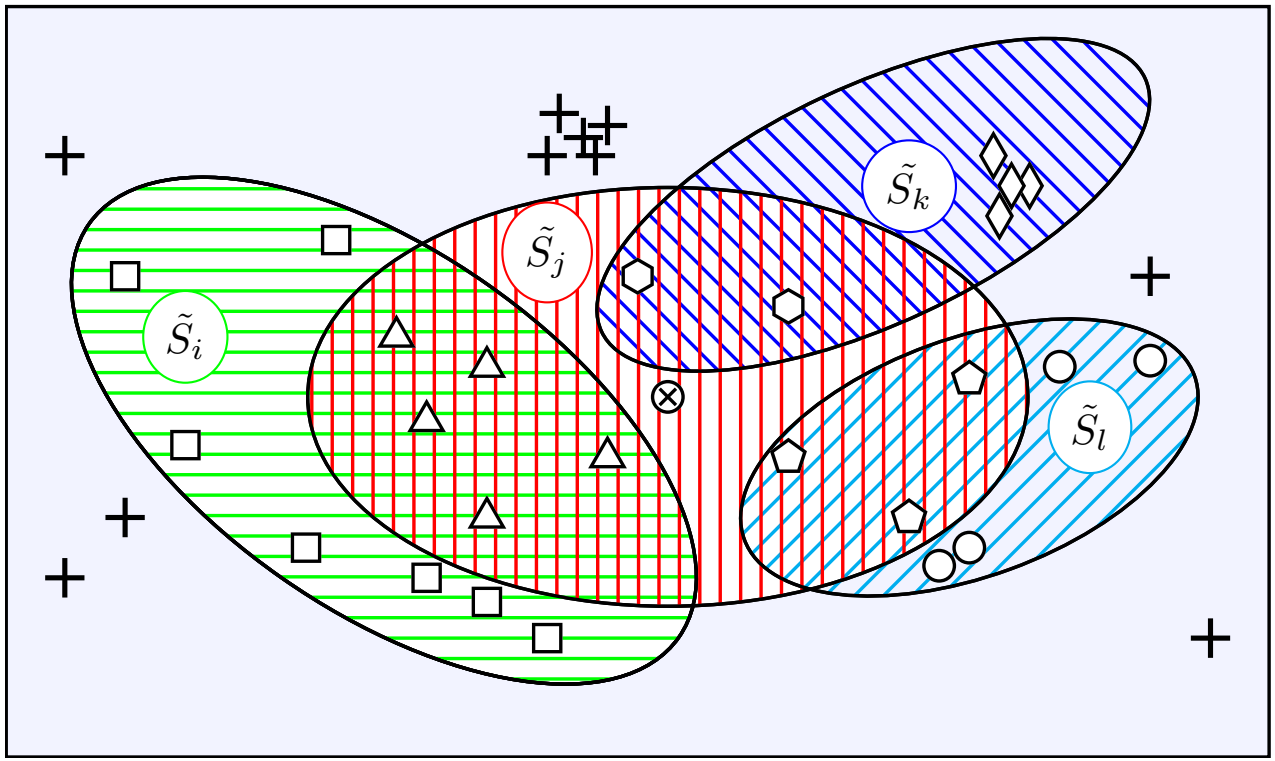
Wichtig

Komposition aus Vereinigung und Teilmengenbeziehung kann nicht zur Signaturanalyse verwendet werden!

Signaturanalyse 2/2

Unbekanntes Verhalten $\otimes$ wird durch $\left[V(\tilde{S}_i) \cup V(\tilde{S}_k) \cup V(\tilde{S}_l)\right]$ nicht erkannt:

$$V(\tilde{S}_j) \not\subset \left[V(\tilde{S}_i) \cup V(\tilde{S}_k) \cup V(\tilde{S}_l)\right]$$



$+ \mathrel{\hat{=}}$ Verhalten
$\square \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_i)$
$\Diamond \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_k)$
$\bigcirc \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_l)$
$\triangle \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_i)$ $\wedge$ Verhalten $\in V(\tilde{S}_j)$
$\hexagon \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_k)$ $\wedge$ Verhalten $\in V(\tilde{S}_j)$
$\pentagon \mathrel{\hat{=}}$ Verhalten $\in V(\tilde{S}_l)$ $\wedge$ Verhalten $\in V(\tilde{S}_j)$
$\otimes \mathrel{\hat{=}}$ unbekanntes Verhalten

Gliederung

Anforderungen und Anwendungen

- Untersuchung verschiedener Anwendungen zur Lösung der Verwaltungsaufgaben; u. A.:
  - Datenbankbasierte Systeme (MySQL, PostgreSQL, VDBS, TDBS)
  - Versionsverwaltungssysteme (SVN, CVS, GIT)
- Unter Beachtung von vorher definierten Anforderungen, wie bspw.
  - Funktionssicherheit
  - Fehlertoleranz
  - Datensicherung
  - Datenspeicherung bzgl. Effizienz
  - ...

Gliederung

Umsetzung und Implementierung

- Konzepte und Ideen von Subversion nutzen
- Datenspeicherung und -verwaltung durch PostgeSQL Datenbanksystem sicherstellen
- Verwaltungsaufgaben „manuell“ implementieren

Konkrete Umsetzung an einem Beispiel

Differentielle Aktualisierung einer Signatur

- Signaturen werden als Strings repräsentiert (EDL-Syntax)
- Globales Bedrohungs-Repository kennt Quell- und Zielliste
- Erstellung einer Differenzenliste
- Übertragung der Differenzenliste an eine EA-Box
- Berechnung der Zielliste in EA-Box aus Quellliste und Differenzenliste

Gliederung

Evaluierung

Differentielle Aktualisierung von Signaturen bzgl. Effizienz

	Quellliste		**Zielliste**	
Daten	Größe (Byte)	# Zeilen	Größe (Byte)	# Zeilen
952 und 961	1.157.958	57.847	78.641	4.062
952 und 962	1.157.958	57.847	82.428	4.252
961 und 962	78.641	4.062	82.428	4.252

	Differenzenliste		**Auswertung**	
Daten	Größe (Byte)	# Zeilen	Zeit (Sek.)	Übertragungsvol.
952 und 961	18.100	302	14,846	0,23
952 und 962	19.347	322	15,027	0,23
961 und 962	64.903	693	0,488	0,79

Gliederung

Zusammenfassung

- Vorstellung der Verwaltungsaufgaben
  1. Verteilung
  2. Versionierung
  3. Revisionierung
  4. Signaturanalyse
- Nennung der Anforderungen an Signaturverwaltungsmechanismen
- Untersuchung von ausgewählten Anwendungen
- Beschreibung der Realisierung
- Evaluierung einer Lösungsstrategie

Ausblick

- PCAP Integration
- Alternativer Ansatz zur Signaturanalyse auf Systemrufebene
- Goodpool-Infrastruktur
- Versionskompatibilität

Gibt es noch unbeantwortete Fragen?

Vielen Dank für die Aufmerksamkeit!