Baccalaureatsarbeit

Analyse der aVTC-Tests
und
Verbesserung der Auswertung

Michel Messerschmidt, 2002

Übersicht

- Überblick über die Tests
- Die alte Auswertungs-Phase
- Die neue Auswertung-Phase
- Allgemeine Kritik
- Fazit
- Wie geht es weiter ?

Überblick über die Tests
Was wird getestet ?

- Es werden Anti-Virus- und Anti-Malware-Produkte getestet
- Keine vollständige Produkt-Evaluation
- Einziges Kriterium: maximal mögliche Erkennungsrate (der vorhandenen Testdaten) eines Produkts
- Es werden (bisher) nur On-Demand-Scanner getestet

Überblick über die Tests
Struktur der Testdaten

- Jedes „testbed“ entspricht einem Laufwerk auf dem Server
- Hierarchische Gliederung nach Malware-Art, -Name und -Variante (realisiert durch Unterverzeichnisse).
  - Beispiel: „H:\W97M\Melissa\A“ enthält Samples von „Melissa.A“
- Testdaten umfassen sowohl virale wie nicht-virale Malware sowie „false positives“

Überblick über die Tests
Ablauf eines Tests

- Jeder Test lässt sich in mehrere Phasen unterteilen:
  - Vorbereitung
    - Produkte empfangen, Rechner und Testdaten vorbereiten
  - Scannen
    - Produkte installieren, Testdaten scannen und Logdateien bereitstellen, Dokumentation des gesamten Vorgangs
  - Auswertung
    - Logdateien der Scan-Phase auswerten und die Ergebnisse für den Testbericht bereitsstellen
  - Testbericht
    - Ergebnisse zusammentragen und bewerten sowie weitere Daten erstellen (z.B. Zeitreihen)

Überblick über die Tests
Auswertung-Aufgaben

- Statische Daten ermitteln
  - Anzahl der Viren & Samples pro testbed
  - Diverse Listen (Dateien, Produkte, ...)
- Verzeichnisse anlegen
- Reportdateien verwalten (Nachscans)
- Produkt-Daten
  - ermitteln (Reportdateien auswerten),
  - **überprüfen (Qualitätssicherung),**
  - zusammenfassen (Tabellen erzeugen)
- Skripte anpassen

Überblick über die Tests
Reportdateien auswerten

- Report in einheitliches Format umwandeln
  - dazu existiert für jedes Produkt ein spezielles 'awk'-Skript („scanner key“)
- Report aufteilen in
  - als infiziert gemeldete Dateien
  - als nicht infiziert gemeldete Dateien
  - alle übrigen Zeilen
- Aus der Liste der infizierten Dateien die Erkennungsrate ermitteln (als absoluter und prozentualer Wert)
  - Anzahl der erkannten Viren
  - Anzahl unzuverlässig erkannter & identifizierter Viren
  - Anzahl der erkannten Samples
- Prüfen, ob alle Dateien gescannt wurden und gegebenenfalls Nachscanlisten erstellen

Die alte Auswertungs-Phase
Vorhandene Skripte

- Es gibt ein Auswertungssystem in Form einer chaotischen Sammlung von Batch-Dateien, 'awk'-Skripten und einigen 'perl'-Skripten die über mehrere Laufwerke verstreut auf dem Server liegen.
- Dieses Auswertungssystem erleichtert im wesentlichen nur das Zählen der Samples und Viren. Aller anderen Aufgaben müssen manuell durchgeführt werden.
- Viele Skripte sind nur historische Überreste und werden nicht mehr verwendet.

Die alte Auswertungs-Phase
Analyse der alten Skripte

Erste Zeile ausgewertet (1 = nein, 42 = ja) ?

Anzahl vorhandener Samples pro Virus

Virusname aus der vorherigen Zeile

Virusname aus der aktuellen Zeile

Pfad + Dateiname der vorherigen Zeile

Unreliable identification (1 = ja, 0 = nein) ?

Diese Variable wird nicht weiter verwendet

Es gilt immer:
i = ANZ
c = 0

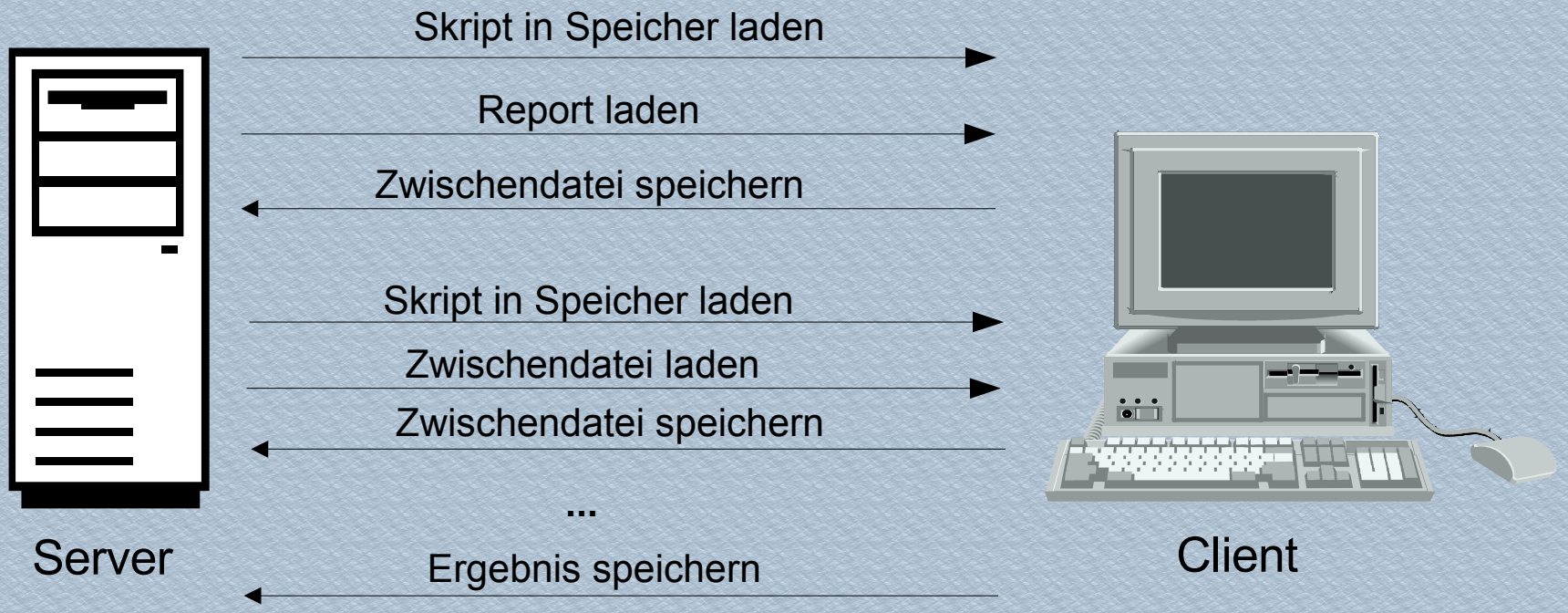
```
# i = exakte Identifikation (Gruppe)
# c = verdacht (Gruppe)
# o = ok (Gruppe)
# ANZ = Anzahl Dateien einer Gruppe
BEGIN{
   scanname = scanner
   aufruf=1
   ANZ=0
   bez=""
   Bezeichnung=""
   Gesamtpfad=""
   gefunden=0
   i=0
   c=0
   o=0
   u=0
         infectsum=0
   scansum = 0
}
```

Die alte Auswertungs-Phase
Randbedingungen

- Es gibt folgenden Einschränkungen:
  - Es gibt keine Selektionsmöglichkeiten - es werden immer alle Reportdateien eines testbeds ausgewertet
  - Nur unter Windows NT oder 2000 lauffähig
  - Benötigt awk, sort, join, wc, perl und arj (z.T. bestimmte Versionen)
  - Einige Skripte benötigen bestimmte Verzeichnisse (D:\temp)
  - Es müssen für jeden Test neue Skripte geschrieben werden, um das Auswertungssystem an die testbeds, Betriebssysteme und Produkte anzupassen
  - Alle anfallenden Daten (auch der Zwischenschritte) werden unkomprimiert auf dem Server abgelegt

Die alte Auswertungs-Phase

Datenaustausch beim Auswerten



Die alte Auswertungs-Phase
Qualitätssicherung

- Von den Skripten wird eine Kontrolldatei generiert, die jedoch nur wenige Daten enthält.
- Alles andere muß per Hand erledigt werden.
- Zur Vereinfachung haben zwar einige Auswerter eigene zusätzliche Skripte geschrieben (z.B. zum Erzeugen der Nachscanlisten), diese sind jedoch mangels Dokumentation nicht mehr nutzbar oder verloren gegangen.

Die neue Auswertungs-Phase
Anforderungen

- Es soll möglichst wenig per Hand durchzuführen sein
- Es muß jederzeit möglich sein, ein Überblick über den Stand der Auswertung zu bekommen
- Möglichst geringer Zeitbedarf und möglichst geringe Serverbelastung
- Alle Abläufe müssen später nachvollziehbar sein
- Es muß möglich sein, sowohl einzelne Reports wie auch alle Reports auf einmal auszuwerten
- Die Skripte sollen einfach bedienbar und gut dokumentiert sein
- Alle Verzeichnis- und Dateinamen müssen leicht anpassbar sein
- Es sollen verschiedene Test parallel ausgewertet werden können
- Die Auswertung soll möglichst plattformunabhängig sein

Die neue Auswertungs-Phase
Konzept

- Es existiert nur noch ein Skript, mit dem sich aber verschiedene Aufgaben erledigen lassen.
- Die verschiedenen Funktionen werden durch Module realisiert.
- Alle Einstellungen werden in einer zentralen Konfigurationsdatei festgelegt.
- Es werden möglichst wenige Hilfsprogramme genutzt.
- Vorteile:
  - einfachere Bedienung
  - leichter erweiterbar
  - übersichtlicher
  - bessere Plattformunabhängigkeit

Zur Implementation wurde „perl“ verwendet

Die neue Auswertungs-Phase
Bedienung

- Syntax: avtctest [-d] [-s] [-c=file] befehl os testbed produkt
- Folgende Befehle existieren:
  - new - Auswertung für neuen Test einrichten
  - update - testbed-Daten aktualisieren
  - eval - Reports auswerten
  - reeval - bereits ausgewertete Reports erneut auswerten
  - archive - Reports nur archivieren und nicht auswerten
  - reset - Status eines Reports ändern bzw. zurücksetzen
  - textresult - Ergebnistabellen erzeugen
  - status - Statusbericht erzeugen
  - (grade - Bewertung durchführen)
  - help - naja, was wohl :-)

Die neue Auswertungs-Phase
Statusinformationen

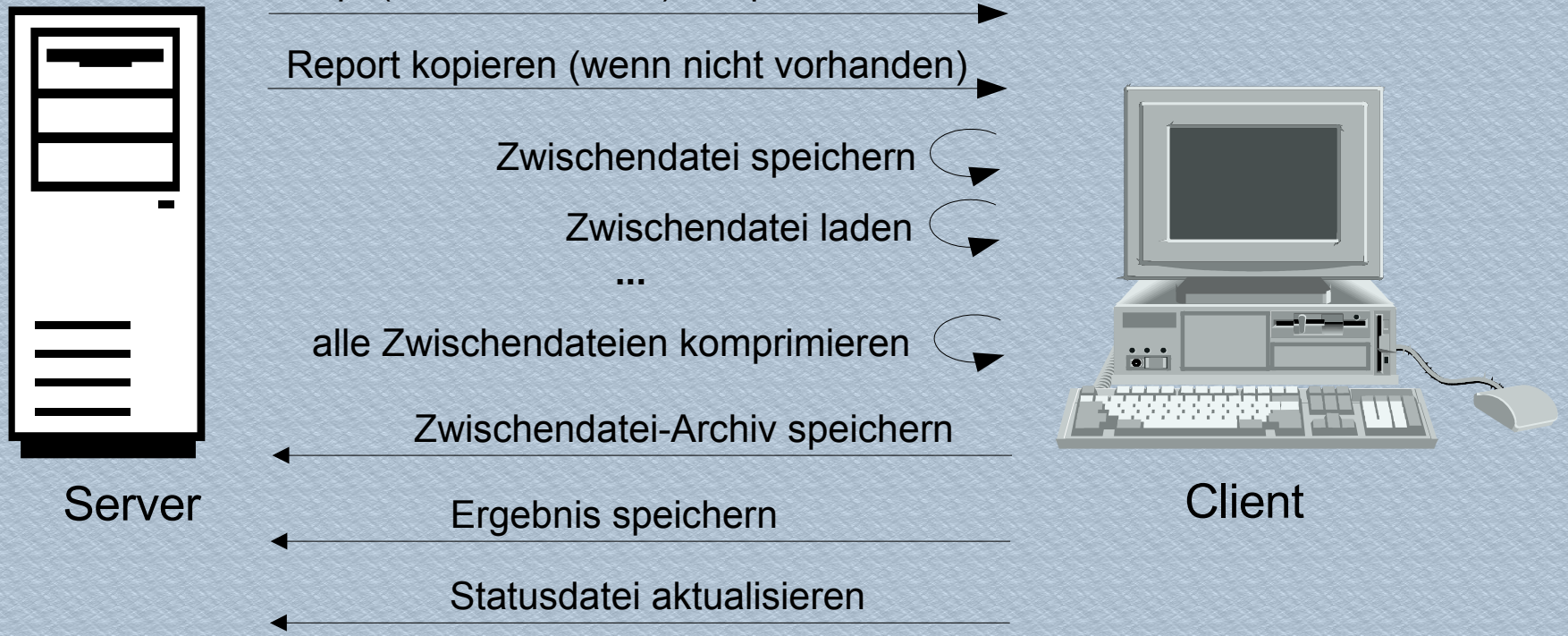
- Zu jeder Reportdatei existiert jetzt ein Statusdatei, aus der der aktuelle Zustand ablesbar ist, z.B.:
  - „NO TEST“ - dieses Produkt wird nicht mit diesem Betriebssystem bzw. testbed getestet
  - „-S1“ - es fehlt der Report des ersten Scans
  - „S1E“ - der Report wurde vom Skript ausgewertet aber noch nicht überprüft
  - „OK S1E“ - Auswertung und Qualitätssicherung erledigt
- Die Notwendigkeit, auf andere Weise über den Testfortschritt Buch zu führen entfällt.
- Es werden jedoch nur für die Auswertung relevante Informationen erfasst, aber z.B. nicht wer welches Produkt testet.

Die neue Auswertungs-Phase
Struktur

- Um skript-gesteuerte Aufgaben auszuführen, müssen alle benötigten Objekte eindeutig identifizierbar sein. Bisher vorhandene Konventionen sind aber nicht eindeutig und z.T. undefiniert.
- Jetzt existieren Listen mit Zuordnungen, z.B.:
  - „J MI_P____ MACR_PAC Packed Macro In-The-Wild“
- Es existieren Listen für Betriebssysteme, testbeds, Produkte und Packprogramme.
- Darauf aufbauend lassen sich skript-gesteuert Verzeichnis-hierachien erzeugen und bearbeiten (z.B. „p:\auswert\2002_04\w2k\macr_pac“).
- Alle für die Auswertung relevanten Dateien liegen jetzt in einer übersichtlichen Struktur, die sich zudem leicht ändern lässt.

Die neue Auswertungs-Phase

Datenaustausch beim Auswerten



Die neue Auswertungs-Phase
Scanner keys

- Mangels Zeit mussten die alten 'awk' scanner keys weiter verwendet werden.
- Nachteil: Der einzige übergebene Parameter ist der Produktname.
- Es kann aber jetzt (optional) pro Betriebssystem ein separater scanner key verwendet werden.
- Es wurde eine neue Schnittstelle für 'perl'-Skripte als scanner keys definiert, bei der alle hilfreichen Informationen übergeben werden können.
- Somit besteht die Möglichkeit, „intelligentere“ scanner keys zu erstellen.

Die neue Auswertungs-Phase
Qualitätssicherung

- Zu fast jedem Zwischenschritt lassen sich Invarianten über die Zeilenzahl der Ein- und Ausgabedateien definieren.
- Alle Tests der Invarianten werden als Gleichung zusammen mit dem Ergebnis der Prüfung in eine Kontrolldatei geschrieben.
- Somit lassen sich Fehler schnell lokalisieren und die Überprüfung des korrekten Ablaufs wird wesentlich einfacher.

Die neue Auswertungs-Phase
Beispiel einer Kontrolldatei

```
                                          lines   words    lines OK ?
---- checks for scanner-key output ----------------------------------------
   original reportfile (DRW.FUL)         :   25695  102837    OK
 = non-evaluated lines (DRW.RES)         :      41     231
 + evaluated lines (DRW.aus)             :   25654  102606
-------------------------------------------------------------------------
   evaluated lines (DRW.AUS)             :   25654  102606    OK
 = scanned but reported as OK (DRW.SBO)  :     515     515
 + unsorted normalized logfile (DRW.NRU) :   25139   50278
---- checks for testbed evaluation ----------------------------------------
   unsorted normalized logfile (DRW.NRU) :   25139   50278    OK
 = sorted normalized logfile (DRW.NRM)   :   25139   50278
-------------------------------------------------------------------------
   normalized logfile (DRW.NRM)          :   25139   50278    OK
 = malware lines evaluated  (DRW.EV1)    :   25110   50220
 + false positive lines (DRW_F.EV1)      :      29      58
 + remaining lines from logfile (DRW.RE1):       0       0
-------------------------------------------------------------------------
   files detected by scanner (DRW.D_F)   :   25139   25139
 + files scanned but ok (DRW.O_F)        :     505     505
 - files infected AND ok (DRW.INO)       :      83      83    check this!
 = all files scanned (DRW.S_F)           :   25561   25561    OK
-------------------------------------------------------------------------
   all files scanned (DRW.S_F)           :   25561   25561
 + files in postscan list (DRW.PSL)      :       0       0
 - all evaluated garbage (DRW.S_G)       :       0       0
 = Files in Testbed (from dirlist)       :   25561   25561    OK
```

Allgemeine Kritik
Vorbereitung

- Das Erstellen der Testdaten ist viel zu aufwändig.
- Die Samples / Testbeds werden (wenn überhaupt) nur von Hand überprüft. Somit bleiben Fehler viel zu leicht unentdeckt.
- Aufgrund der heterogenen Hardware ist das Neuinstallieren der Rechner ziemlich umständlich. Einheitliche Hardware könnte dies wesentlich vereinfachen.

Allgemeine Kritik
Auswertung

- Bisher wird nur nach „**infiziert**“ und „**nicht infiziert**“ unterschieden. Die Meldungen der Produkte sind aber nicht immer so eindeutig (insbesondere bei Malware):
  - „Possibly infected with an unknown virus“
  - „Could be a destructive program“
  - „suspicious code found“
  - „suspect: Macro.VBA“
  - „Is a joke program“
  - „contains macros“
- Vorschlag: neue Kategorien einführen, um nicht-virale Malware besser zu berücksichtigen:
  - „**viral**“ / „**schädlich**“ / „**harmlos**“

Allgemeine Kritik
Testbericht

- Der Testbericht ist viel zu umfangreich (z.T. über 200 Seiten).
- Durch ein besseres Format als ASCII-Text (z.B. HTML) lässt sich die Lesbarkeit wesentlich verbessern.
- Der Testbericht enthält zu viele „copy & paste“-Fehler.
- Einige Teile des Testberichts müssen nicht per Hand erstellt werden, sondern könnten auch skript-gesteuert erzeugt werden (z.B. die Bewertung).
- Alle Inhalte sollten auf Ihre Aussagekraft überprüft werden.

Allgemeine Kritik
Testbericht - Beispiel

Table W2k.M4: "False Positive" macro virus detection: Results of "full" zoo test for non-viral (clean) macro objects detected as "false positives" under Windows-2000:

Scanner	False Virus Alarm		This identified	includes unreliably detected		Files detected	
Maximum	26	100.0%				329	100.0%
AVK	0	0.0	0	0.0	0	0	0.0
AVP	2	7.7	0	0.0	2	4	1.2
CMD	1	3.8	0	0.0	1	2	0.6
DRW	10	38.5	0	0.0	10	29	8.8
INO	0	0.0	0	0.0	0	0	0.0
MCV	26	100.0	0	0.0	26	118	35.9
NAV	5	19.2	0	0.0	5	5	1.5
NVC	3	11.5	0	0.0	3	5	1.5

Fazit

- Alle Anforderungen wurden erfüllt.
- Die Auswertung ist deutlich einfacher und schneller geworden.
- Es ist jetzt möglich, sich wieder auf wichtigere Dinge zu konzentrieren als die reine Ergebnisermittelung.

Wie geht es weiter ?

- Verwaltung der Viren-Kollektionen duch eine Datenbank (oder andere Methoden, die die Verwaltung vereinfachen)
- Überarbeitung der Testberichte
- Neue Tests bzw. Testmethoden