RUHR-UNIVERSITÄT BOCHUM

RUB

# Erkennung von bösartigen Netzwerkverbindungen mittels Verhaltensgraphenanalyse

hgi
Horst Görtz Institut
für IT-Sicherheit

Ralf Hund

**21.03.2011 - SPRING 6**

**Arbeitsgruppe Embedded Malware**
Lehrstuhl für Netz- und Datensicherheit

# Zur Person

- Diplom an der Universität Mannheim im Jahr 2009
- 2009-2010: Wiss. Mitarbeiter am Lehrstuhl Praktische Informatik I bei Prof. Felix Freiling an der Universität Mannheim
- Seit 2010: Wiss. Mitarbeiter in der Arbeitsgruppe *Embedded Malware* bei Prof. Thorsten Holz an der RUB

- Forschungsinteressen:
  - Malwareanalyse (dynamisch und statisch)
  - Softwareschwachstellen / Neue Angriffstechniken

# Einleitung

- Betrachten im Folgenden ***Bots***
- Seit Jahren existierendes, weitgehend ungelöstes Problem im Internet
- Unterscheidungsmerkmal zu anderer Malware: Aufbau eines *Rückkanals* zu einem Kontrollserver (C&C Server)
- C&C Server gibt neue Befehle und empfängt gesammelte Daten

# C&C Verbindungen

- Typische erteilte Befehle sind u.A.:
  - *Updatefunktion*: Installiere neue Botversion von http://...
  - Denial of Service
  - Versand von *Spam*
  - *Weiterverbreitung* mittels Exploits
- Gesammelte Daten umfassen:
  - Lokal gespeicherte *Zugangsdaten*
  - Gesniffte Netzwerkdaten

# Problemstellung
## Identifizierung von C&C Verbindungen

- *Identifizierung* von C&C Verbindungen notwendig um nähere Informationen über das zugrundeliegende *Botnet* zu erhalten
- Generierung von *Blacklists* (IPs von C&C Servern)

- ~ 40% aller Netzwerkverbindungen von Bots sind *kein* C&C Traffic
- Nicht jede Bot-Verbindung ist per se „böse“
  - Viele Bots bauen (absichtlich) *normale* Verbindungen auf
  - Verfügbarkeitschecks, NTP-Server, Webseiten, etc.

# Identifizierung von C&C Verbindungen
## Ansätze

- Traditionell: *Netzwerkbasierter* Ansatz
- Überprüfung der Netzwerkdumps nach Ausführung des Samples
  - Pattern-Matching nach bestimmten Strings
- Problem:
  - Ungenau
  - Verschlüsselung

# Hostbasierter Ansatz

- Verknüpfung der *Host-Informationen* aus Sandbox mit Netzwerkdaten
  - „Bessere“ Informationen

- *Systemaufrufe* (system calls): API Funktionen mit Parameterwerten
- *Tainting*-Informationen: Wie werden Daten im Programm weiterverarbeitet?

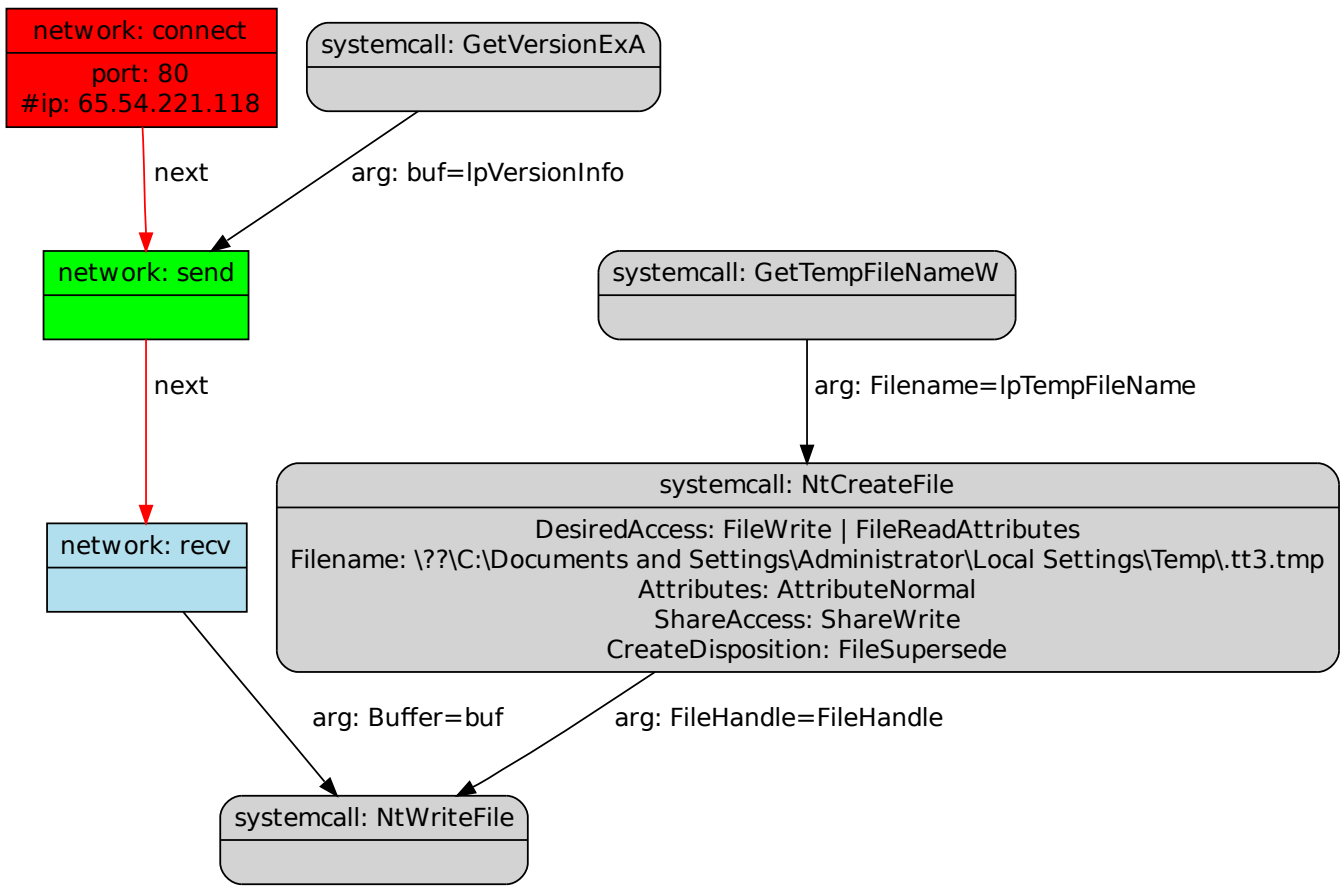
# Automatisierte Analyse
## Beispiel Anubis

- Zugriff auf Anubis Livesystem (http://anubis.iseclab.org)
- Anubis: Sandbox-System basierend auf QEMU
- Jedes Sample wird 3 Minuten in kontrollierter Umgebung ausgeführt und ausgeführte Operationen werden protokolliert
- Täglich ca. 80.000 Analysen

# Verhaltensgraphen

- Darstellung beider Informationen in einem *Verhaltensgraphen*
  - Pro Netzwerkverbindung ein Verhaltensgraph
  - *Knoten*: Systemaufruf
  - *Kante*: (Taint-)Beziehung zwischen zwei Aufrufen

# Verhaltensgraph
## Beispiel



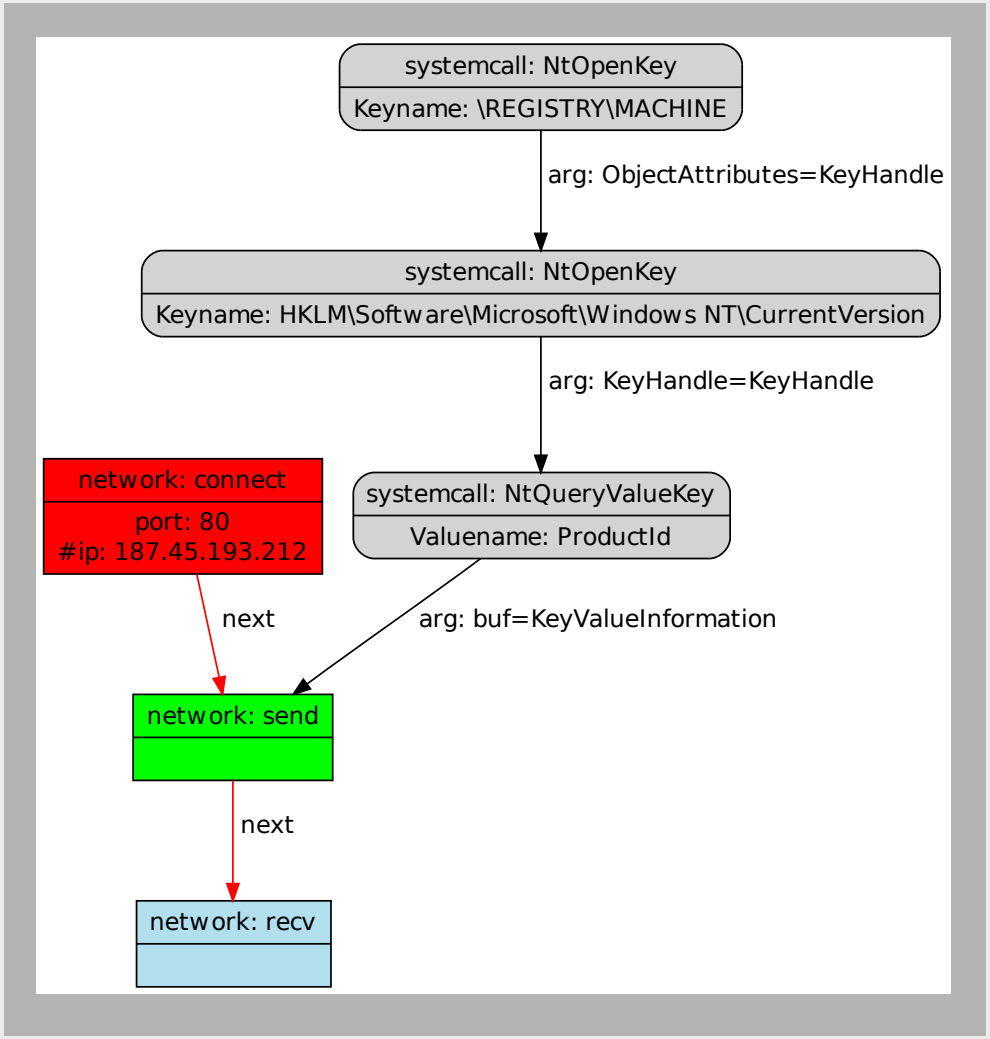
# Merkmale C&C Verbindungen

- Was sind die herausragenden Merkmale der Verhaltensgraphen von C&C Verbindungen?
- *Updatefunktion*: Über das Netzwerk empfangene Daten werden auf Festplatte geschrieben und ausgeführt.
- *Information Leakage*: Rückgabewerte bestimmter Funktionen werden versendet
- *Proxyverhalten:* Daten aus anderer Verbindung werden versendet

# Information Leakage
## Beispiel

- Senden der Windows ProductId
- Request:

```
GET /trade/imgs/doit.php?
v=3&id=2c632d8a-76487-640-1457236-23837 HTTP/1.1
x-type: promake
User-Agent: Mozilla/4.0 (compatible; MSIE 6.1;
Windows XP)
Host: www.depositodevideos.com.br
```



# Erkennungsansätze

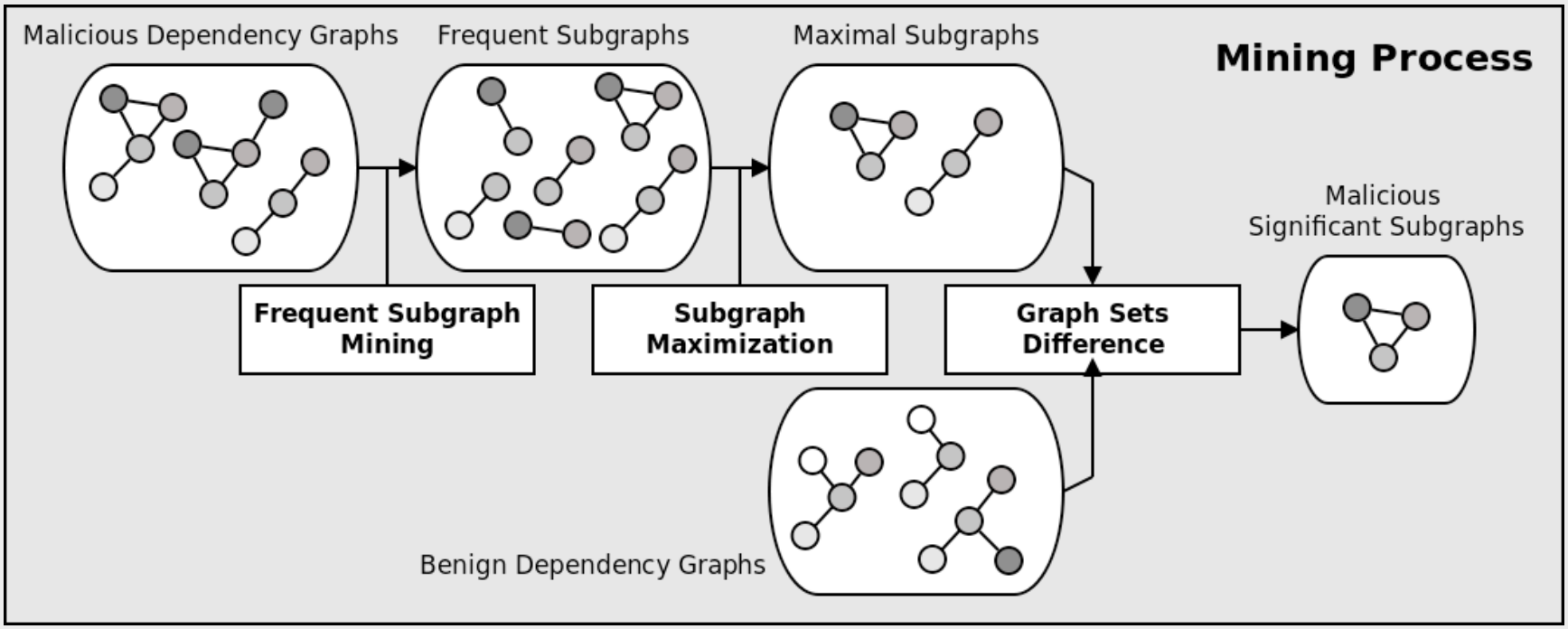
- „Manueller“ Ansatz
  - Erkennen von typischen Eigenschaften im Verhaltensgraphen
  - Plus: Präzise Ergebnisse
  - Minus: Weiterer „human“-input

- „Data Mining“ Ansatz
  - Automatisierte Generierung von *Templates*
  - Templates werden zur Erkennung verwendet

# Erkennung mittels Data Mining

- *Gegeben*: Menge von bekannten gutartigen und bösartigen Verbindungen (Trainingsmenge)
  - Ermittelt über klassische Netzwerksignaturen
- *Ziel*: Templates für bösartige Verbindungen
- Bestehen aus Kern-Knoten und optionalen Knoten

# Data Mining


Malicious Dependency Graphs
Frequent Subgraphs
Maximal Subgraphs
Mining Process
Malicious
Significant Subgraphs
Frequent Subgraph Mining
Subgraph Maximization
Graph Sets Difference
Benign Dependency Graphs

# Evaluation
## Setup

- ~ 37.000 einzigartige Samples
  - ~ 130.000 Netzwerkverbindungen wurden erzeugt
- Netzwerksignaturen
  - jeweils ~ 16.000 C&C / gutartige Verbindungen
- Unterteilung in Trainings- und Testmenge

# Evaluation
## Ergebnisse Testmenge

- 81% der C&C Verbindungen in der Testmenge erkannt
- Gründe für False Negatives
  - Verhalten nicht vollständig (Timeout)
  - Grauzone zwischen gut und böse (Adware)
  - Verhalten teilweise zu selten

# Evaluation
## Ergebnisse Unknown-Menge

- Unknown-Menge besteht zu großen Teil aus fehlerhaften Verbindungen (HTTP 404, keine Antwort, etc.)
- ~ 60.000 sinnvolle Verbindungen (60%)
- 15% Treffer (~ 9.000)
- 200 neue Malwarefamilien, die von Netzwerksignaturen nicht erfasst werden

- Future Work: Vergleich zu *manuellen* Templates

# Danke für Ihre Aufmerksamkeit!

In Zusammenarbeit mit Gregoire Jakob, Thorsten Holz und Christopher Kruegel
Eingereicht bei USENIX Security 2011