

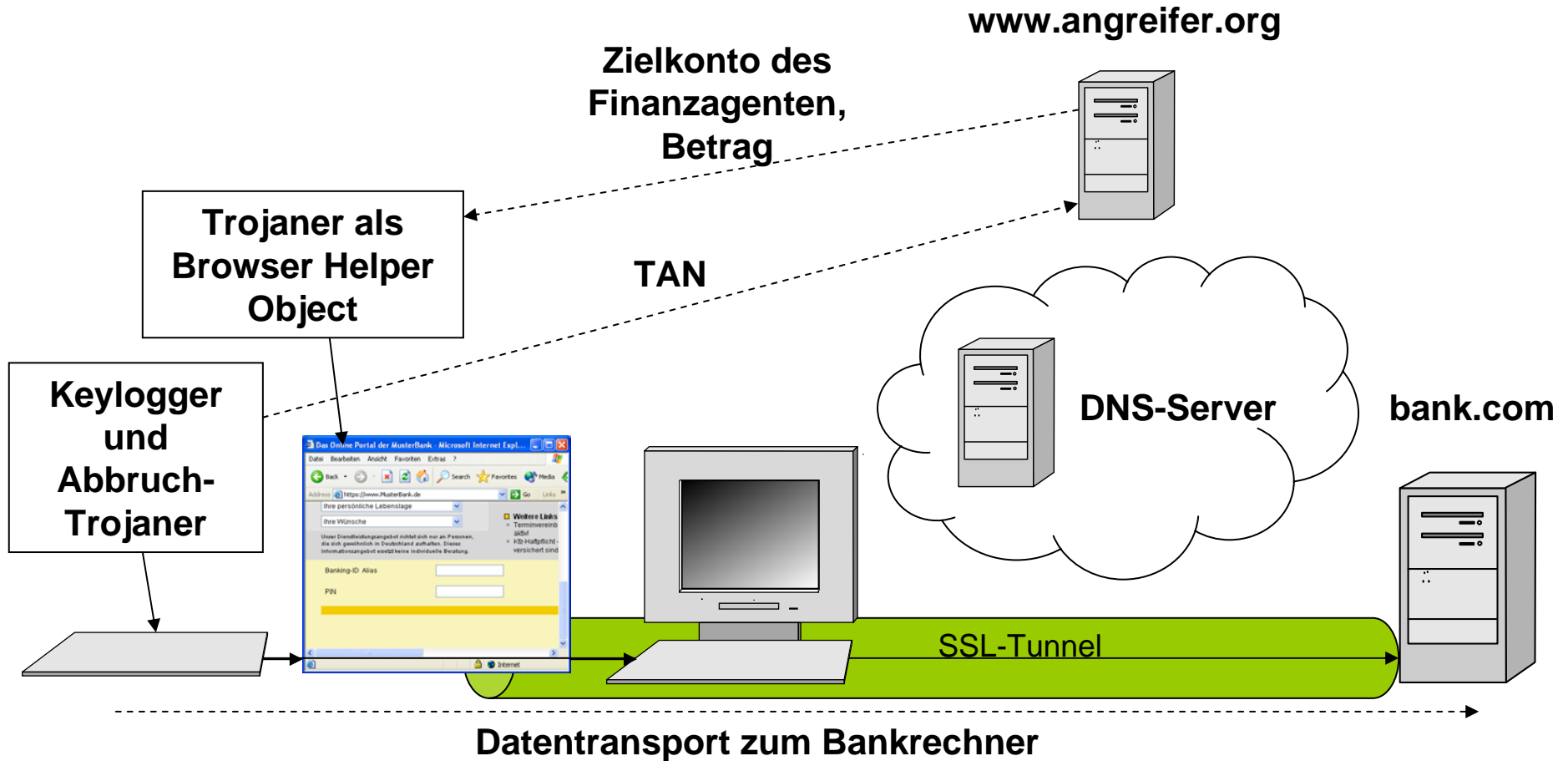
1. Preis: Thomas Dullien, Sabre Security

VxClass – Automatische
Klassifikation von Malware und
Trojanern in „Familien“

VxClass – Die Problemstellung

- Bot, Virus, Worm, Trojan, Keylogger, Rootkit, Spyware,...
- Malware gestern:
 - technisch versierte Hacker
 - Code in Assembler/Maschinensprache
 - Ziel: „Ruhm“ durch hohen Verbreitungsgrad
- Malware heute:
 - Toolkits zur Unterstützung der Programmierung
 - Code in Hochsprache geschrieben
 - verschiedenste, auch mehrstufige Infektionswege
 - geringer Verbreitungsgrad, viele spezialisierte Varianten
 - Ziel: Finanzieller Gewinn

VxClass – Die Problemstellung



VxClass – Die Problemstellung

- SSL-Trojaner: Roger A. Grimes „An SSL Trojan unmasked“ (www.infoworld.com)
 - Trojaner durchsucht die „Temporary Internet Files“ um herauszufinden, welche Bankseiten das Opfer besucht.
 - Er legt eine lokale Kopie der Login-Seite dieser Bank an.
 - Er fängt den Aufruf der echten Login-Seite ab, und lädt die lokale Seite. Das SSL-Schloss bleibt unverändert.
 - Die eingegebene PIN wird an den Angreifer und an die „echte“ Login-Seite gesendet.

VxClass – Die Problemstellung

- Malware wird „auf Kundenwunsch“ maßgeschneidert hergestellt (z.B. mit Schadfunktion nur gegen bestimmte spanische Banken)
- Verbreitungsgrad ist klein
- Anti-Virus-Signaturen passen nicht mehr
- Manuelle Analyse der vielen Varianten stößt an ihre Grenzen
- Lösungsmöglichkeiten:
 - Verhaltensanalyse (Behavioral Analysis)
 - Codeanalyse/Strukturanalyse (VxClass)

VxClass – Alternative: Verhaltensanalyse

- Malware wird in virtueller Umgebung ausgeführt, die Aktionen werden protokolliert
 - Norman SandBox (Norman ASA)
 - TTAlyze (Ulrich Bayer)
 - sandnet (Chas Tomlin)
 - Truman - The Reusable Unknown Malware Analysis Net (LURHQ)
 - CWSandbox (Carsten Willems)
- Protokollierung der Änderungen an Windows Registry, Dateisystem, Netzwerkverkehr, ...

VxClass – Alternative: Verhaltensanalyse

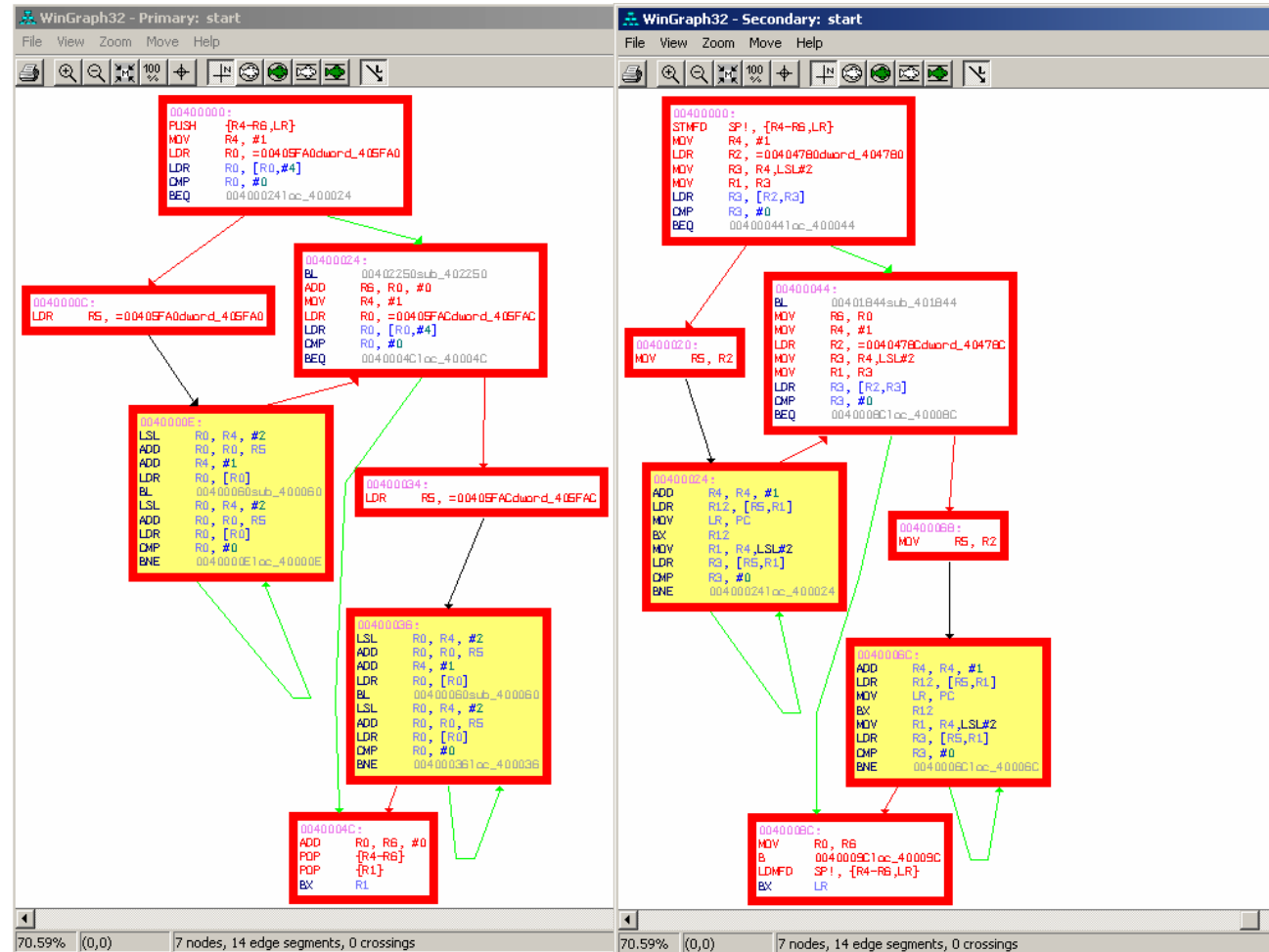
```
<?xml version="1.0"?>
<!-- This analysis was created by the CWSandbox (c) Carsten Willems 2006-->
<analysis><calltree>...</calltree><processes><process ...>
<virusscan_section>...</virusscan_section>
<default_section>...</default_section>
<dll_handling_section>
<load_dll dll="C:\WINDOWS\system32\ntvdm.exe" successful="1"/> ...
<load_dll dll="uxtheme.dll" successful="1"/>
</dll_handling_section>
<filesystem_section>
<get_file_attributes filetype="File" srcfile="C:\WINDOWS\_default.pif" desiredaccess="FILE_ANY_ACCESS"
  flags="SECURITY_ANONYMOUS" fileinformationclass="FileBasicInformation"/> ...
<create_open_file filetype="File" srcfile="C:\WINDOWS\TEMP\scsF3.tmp" creationdistribution="OPEN_ALWAYS"
  desiredaccess="FILE_ANY_ACCESS" shareaccess="SHARE_READ,SHARE_WRITE"
  flags="FILE_ATTRIBUTE_TEMPORARY,SECURITY_ANONYMOUS"
  fileinformationclass="FileBasicInformation"/>
</filesystem_section>
<registry_section>
<query_value key="HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System" subkey_or_value="Identifier"/>
...
<query_value key="Control Panel\Desktop" subkey_or_value="LameButtonText"/>
</registry_section>
...
</process></processes></analysis>
```

VxClass – Strukturanalyse

- Vorbereitung: Entpacken und Dissassemblieren
- 1. Stufe: Erstellen von Funktionsgraphen
 - Identifikation der Sprünge in einem Codeblock: jeder Sprung wird zu einer gerichteten Kante
 - Jeder Funktionsgraph hat eindeutigen Einstiegspunkt
- 2. Stufe: Erstellen des Callgraphen
 - Funktionsgraphen sind durch Funktionsaufrufe untereinander verbunden: jeder Aufruf wird zu einer gerichteten Kante
- 3. Stufe: Vergleich des Callgraphen mit bereits bekannten Callgraphen
- 4. Stufe: Vergleich der Funktionsgraphen, die an ähnlicher Stelle sitzen

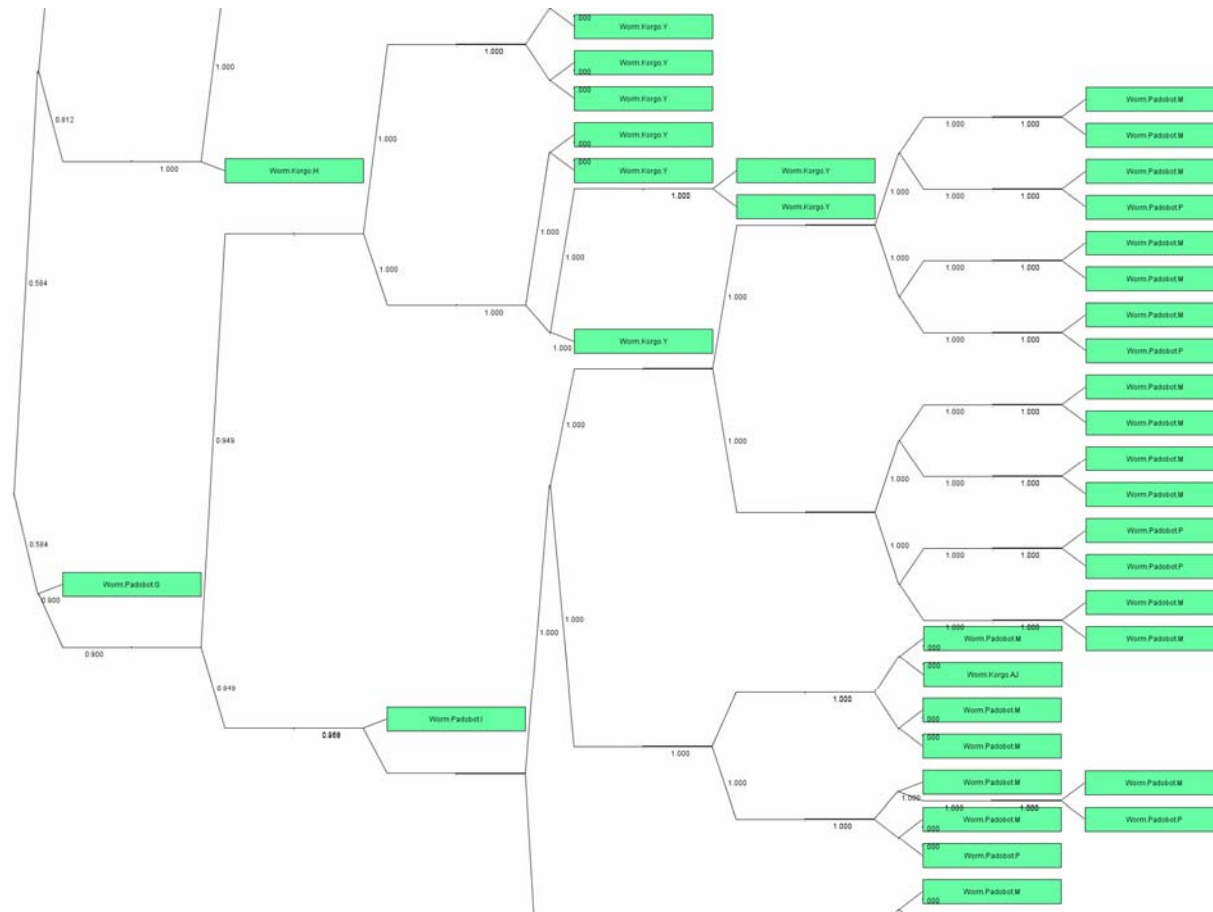
VxClass – Funktionsgraph

Commwarrior.A
vs.
Commwarrior.C



VxClass – Strukturanalyse

Algorithmen aus der Bioinformatik: Erzeugung von Stammbäumen mittels Ähnlichkeitswerten



VxClass – Nutzen

- Asymmetrie im Arbeitsaufwand zwischen dem Erstellen von Malware-Varianten und ihrer Analyse.
- Diese Asymmetrie kann durch den Einsatz von VxClass deutlich verringert werden.
 - Beispiel: Durch Vergleich von Bagle/X mit einer bereits analysierten version von Bagle/W konnten automatisch 223 von 236 Funktionen identifiziert werden.
- Für jede Malware-Familie muss nur noch ein Mitglied zeitaufwändig analysiert werden.
- Die Namensgebung für Malware kann vereinheitlicht werden.