Rethinking Antivirus:
Executable Analysis in the Network Cloud

Jon Oberheide, Evan Cooke, Farnam Jahanian
University of Michigan

August 7, 2007

HOTSEC '07

Limitations of Antivirus

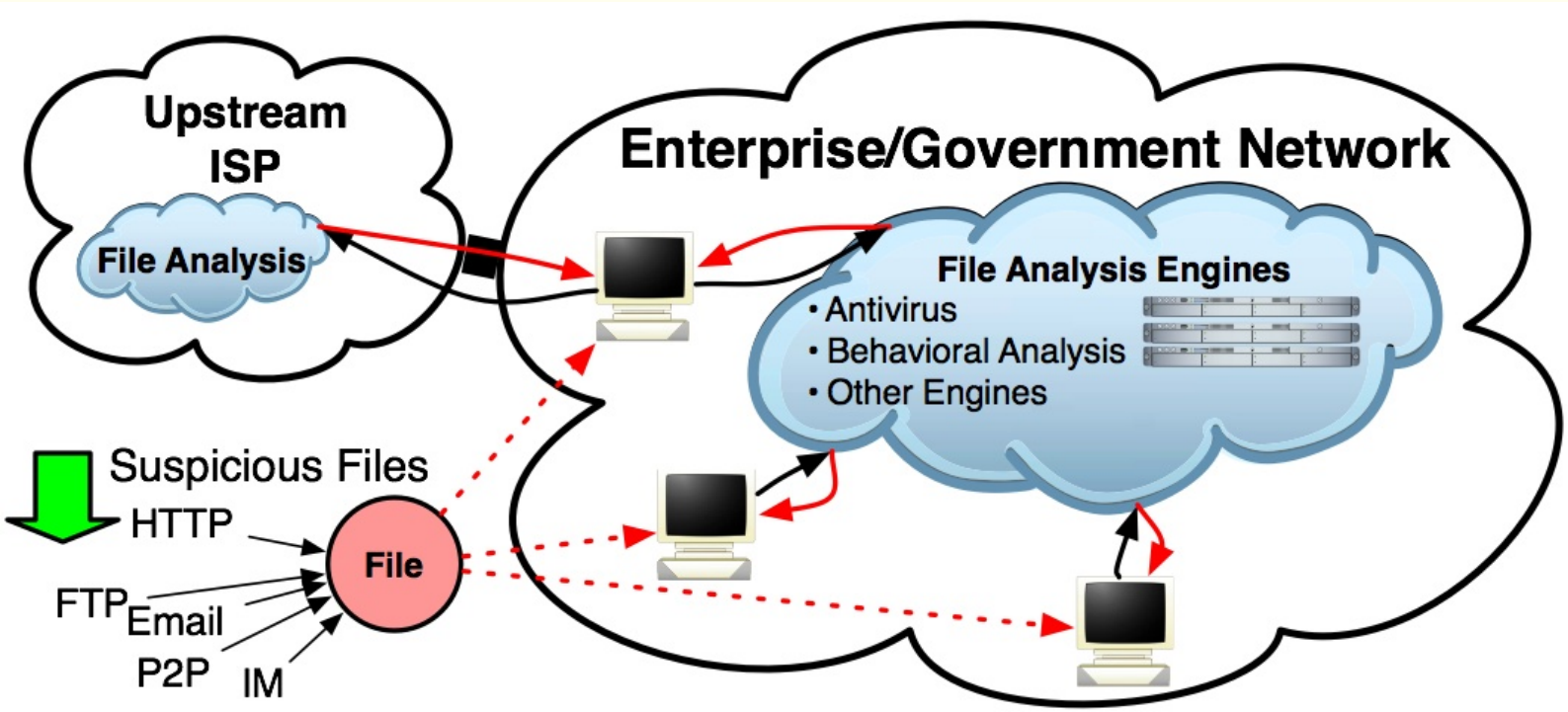
Antivirus is the predominate method of detecting and stopping malicious software

- AV fails to detect modern threats
  - Worst: 54.9% Best: 86.6% Avg: ~76%
- AV host software is complex
  - Maintenance overhead
    - Frequent signature updates
  - Risk of security vulnerabilities
    - Rinbot – Symantec remote exploit

Antivirus	Detected
Avast	**84.7%**
ClamAV	**59.7%**
F-Prot	**79.9%**
F-Secure	**86.6%**
Kaspersky	**85.3%**
McAfee	**54.9%**
Symantec	**81.9%**
Trend Micro	**82.0%**

AML dataset of 5066 samples (Sept '06 – May '07)

AV as a In-Cloud Network Service



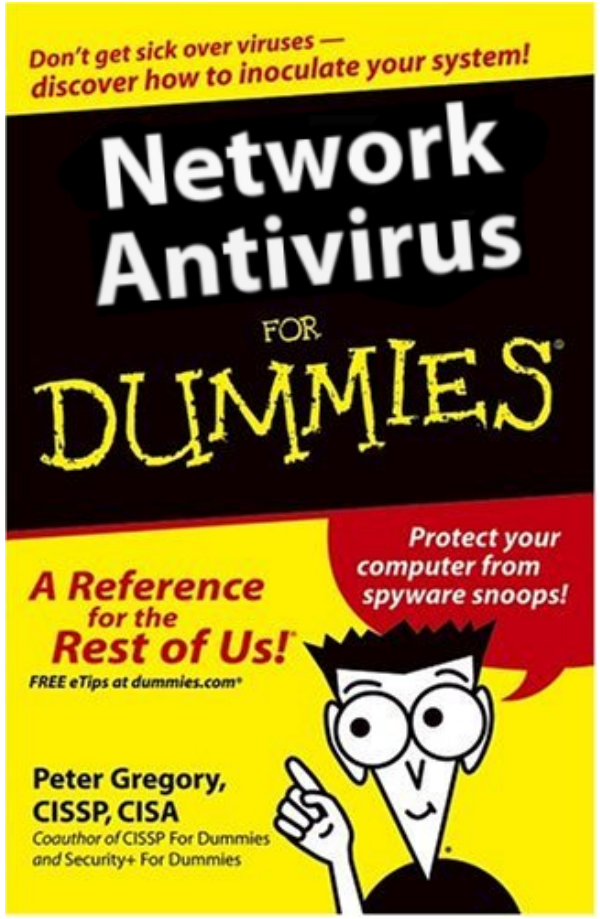
- Attributes of an in-cloud antivirus network service
  - Parallel analysis with multiple detection engines
  - Simplified host agent software
  - Centralized management and network-wide visibility
  - Information sharing between detection engines

More is Better

- Multiple detection engines
  - Parallel, scalable analysis
  - Heterogeneous engines
  - Increased detection coverage
- Detection engine classes
  - Antivirus products
  - Behavioral simulators
  - Other detectors
    - Static and dynamic analysis
    - Easily extended for new engines

Keep It Simple Stupid

- Simplified host agent software
  - Eliminate frequent updates
  - Mobile and other resource-constrained devices
  - Reduce vulnerability profile
- Centralized management
  - Network-wide visibility
    - Malicious threats
    - Legitimate executable usage



Sharing is Caring

- Detection engines can share info
  - Correlation enables greater detection coverage
  - Caching mechanisms enable performance enhancements
- Example scenario
  - Malicious executable not detected by antivirus engines
  - Behavioral engine finds behavior identical to a previous executable detected by antivirus
  - Executable flagged as malicious

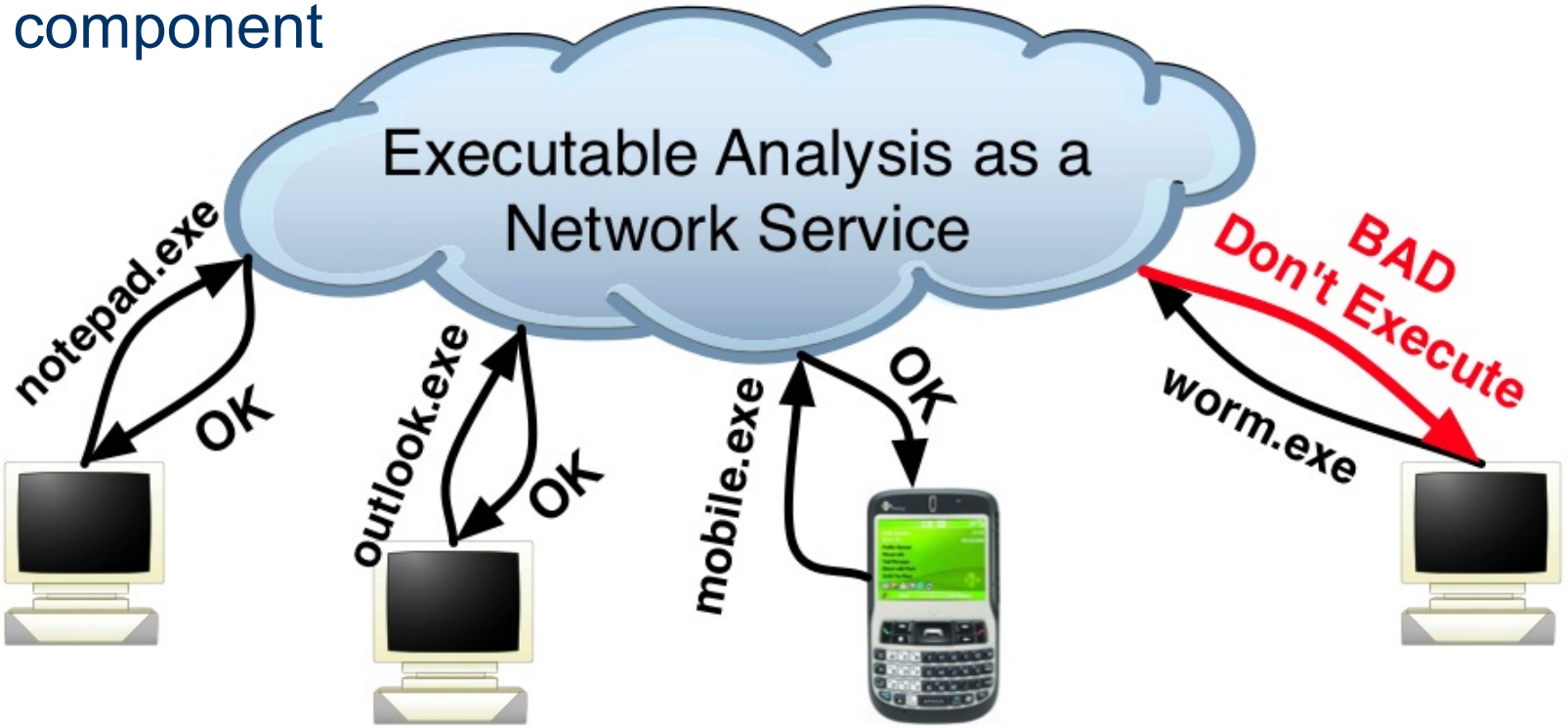
Share Bear says...

Sharing is caring!

Stop that malware!

Implementation: Overview

Network service component



Host agent component

Implementation: Overview

- The *host agent* detects a new *executable* on host
- The *host agent* blocks access to the *executable*
- *Executable* checked against local black/white lists
- The *host agent* sends hash of the *executable* to the *network service*
- The *network service* checks the hash against its black/white lists
- The *host agent* sends the *executable* to the *network service* for analysis
- The *network service* analyzes the *executable* and sends a threat report back to the *host agent*

Implementation: Host Agent

- Win32 Implementation
  - Local black/white list cache
  - File system notifications
  - CreateProcess API hooking
- Disconnected operation
  - Mobile, DoS, outage
  - Policy decision
    - Fail-over local AV

Implementation: Network Service

- In-cloud network service
  - Virtualized detection engines
    - VMware-based containers
    - Scalability and security
  - Remote black/white list caching
  - Multiple engines running in cluster
    - Antivirus products
    - Behavioral engines
      - Norman Sandbox Analyzer
      - VMware/VTrace Profiler
    - Extensible to new engines

Initial Results: Coverage

#	Antivirus Products Run in Parallel	Detected
1	F-Secure	**86.59%**
2	Trend, Avast	**92.93%**
3	Trend, F-Secure, Avast	**94.63%**
4	ClamAV, Symantec, Trend, Avast	**95.34%**
5	ClamAV, Symantec, Trend, F-Secure, Avast	**95.85%**
6	F-Prot, ClamAV, Symantec, Trend, F-Secure, Avast	**96.15%**
7	Mcafee, F-Prot, ClamAV, Symantec, Trend, Kaspersky, Avast	**96.23%**
8	Mcafee, F-Prot, ClamAV, Symantec, Trend, F-Secure, Kaspersky, Avast	**96.23%**

- Multiple AV engines detect 4875 of 5066, 191 undetected
- Correlation with behavioral analysis
  - 92 of 191 have identical behavior to known malicious samples

Total detection coverage of over 98%

Initial Results: Performance

- Local Network Assumptions
  - Low-latency (<100ms)
  - High-speed (>=100Mbps)
- Analysis times of legitimate and malicious samples:

Legitimate Dataset	
Executables	**472**
Avg Size	**183 KB**
Avg AV Time	**0.05s**
Worst AV Avg	**0.14s**

Malicious Dataset	
Executables	**5066**
Avg Size	**366 KB**
Avg AV Time	**0.48s**
Worst AV Avg	**0.91s**

Reasonable analysis times (<1 second)

Initial Results: Caching

- Are black/white list caches an effective optimization to eliminate redundant analysis?

- Dataset from mwcollect Alliance
  - /18 network, 2 month period
  - 213 distinct executables
    - seen over 2.5 million times
    - 49 seen once, 164 seen multiple

Preliminary results indicate effective hit rates

Wrap-up

- Initial prototype feasibility
  - Coverage: increased!
    - Over 98% in dataset
  - Performance: acceptable!
    - Potentially increased perf w/caching
  - Deployability: positive!
    - Better management/visibility
- Future developments and evaluation
  - Production-grade implementation
  - University deployment (~1k hosts)
- In-cloud network security service
  - enterprise cloud, organizational networks, upstream ISP

Questions

QUESTIONS?