Why Protection against Viruses, Bots, and Worms is so hard
Malware seen as Mobile Agents

Till Dörges
td@pre-secure.de

PRESECURE Consulting GmbH

June 20, 2007
Table of Contents

1 Foundations
   Agents and Multi Agent Systems
   Agents and Malware

2 Security in MAS
   Desirable Properties
   Protecting the Platform
   Protecting the Agent

3 Conclusion
1 Foundations
   Agents and Multi Agent Systems
   Agents and Malware

2 Security in MAS
   Desirable Properties
   Protecting the Platform
   Protecting the Agent

3 Conclusion
# Table of Contents

1. **Foundations**
   - Agents and Multi Agent Systems
   - Agents and Malware

2. **Security in MAS**
   - Desirable Properties
   - Protecting the Platform
   - Protecting the Agent

3. **Conclusion**
Agents

What is an Agent?
Agents

What is an Agent?

- Modeling Paradigm
  - Software Engineering (unlike e.g. objects, ...)
  - Artificial Intelligence
Agents

What is an Agent?

- Modeling Paradigm
  - Software Engineering (unlike e.g. objects, ...)
  - Artificial Intelligence

Important Properties

- Encapsulation and Modularization
- Reactivity
- Proactivity
- Autonomy
- Mobility (not generally required)
Agents (cont’d)

Definition

- Subject to quite a bit of debate
  - Social Behavior
  - Ability to Adapt
  - Goal Orientation
  - …

- Key properties are safe to assume

Particularly Suited for

- Distributed and Concurrent Systems
- Systems across Multiple Administrative Domains
Agents (cont’d)

Colloquially Speaking

- Program/Code and Data
- Travel between Platforms
- Run on different Platforms

Examples

- “Shopping Agent”
  - “Find (buy) a blue Bicycle for not more than EUR 500.”
  - Inquires at several platforms
  - Finds best solution
  - Possibly purchases a bike on behalf of owner/user
Distinction from Mobile Code

Examples for Mobile Code

- JAVA applets
- ActiveX controls
- ...

Mobile Code lacks

- Autonomy
- Proactivity
- Goal Orientation
Platforms

What is a Platform?

- Runtime Environment for Agents
- Responsible Protection of Agents
- Services for Interaction (communication, directory services, ...)
- Transportation of Agents between Platforms

Colloquially Speaking

- Application on a Computer
Multi Agent Systems – MAS

What is a MAS?

- Technically
  - $n$ with $n > 0$ Platforms
  - $m$ with $m > 0$ Agents
  - Infrastructure/Policies
- Service Point of View
  - Shopping Platform
  - Database Querying
  - Research
  - …
- Multi Agent Application
  - …
Agent Orientation as Modeling Paradigm

- Comparable to Object Orientation
- AO development environments readily available
- AO application doesn’t have to show agents on the outside
Table of Contents

1 Foundations
   Agents and Multi Agent Systems
   Agents and Malware

2 Security in MAS
   Desirable Properties
   Protecting the Platform
   Protecting the Agent

3 Conclusion
Malware

Definition (Wikipedia)

Malware is software designed to infiltrate or damage a computer system without the owner’s informed consent. . . . [The term designates] a variety of forms of hostile, intrusive, or annoying software or program code.

Taxonomy

- Species
  - Virus
  - Bot
  - Worm
  - . . .
- Distinction blurry
Malware (cont’d)

Properties

• Provision of “Services”
  • Spying
  • Attacking
  • Back Doors
  • …

• Reactivity
• Proactivity
• Autonomy
• Mobility
• Self Replication
• Adaption
Malware (cont’d)

Properties

• Provision of “Services”
  • Spying
  • Attacking
  • Back Doors
  • …

• Reactivity
• Proactiveness
• Autonomy
• Mobility
• Self Replication
• Adaption
Comparison

Malware?
- Comparison Malware $\Leftrightarrow$ Agents holds

Platforms?
- Infected Computers provide for Runtime Environment
- Other services implemented by Malware directly
- Comparison for Infected Computers $\Leftrightarrow$ Platforms holds

MAS?
- Less interesting (1 malware is enough to control 1 computer)
- Holds, too.
Comparison

Malware?

- Comparison Malware ⇔ Agents holds

Platforms?

- Infected Computers provide for Runtime Environment
- Other services implemented by Malware directly
- Comparison for Infected Computers ⇔ Platforms holds

MAS?

- Less interesting (1 malware is enough to control 1 computer)
- Holds, too.
# Table of Contents

1. **Foundations**
   - Agents and Multi Agent Systems
   - Agents and Malware

2. **Security in MAS**
   - Desirable Properties
   - Protecting the Platform
   - Protecting the Agent

3. **Conclusion**
## Table of Contents

1. Foundations
   - Agents and Multi Agent Systems
   - Agents and Malware

2. Security in MAS
   - Desirable Properties
   - Protecting the Platform
   - Protecting the Agent

3. Conclusion
Security

Conventional Aspects / Definition

- Confidentiality
- Integrity
- Availability
Security

Conventional Aspects / Definition

- Confidentiality
- Integrity
- Availability
Security

Conventional Aspects / Definition

- Confidentiality
- Integrity
- Availability

Shortcomings

- Every System is Special
- Definition has to be adapted
- What about (for example)
  - Identity
  - Trust
  - …
Desirable Security Properties in MAS

Security for Agents?

- Communication
  - Integrity
  - Confidentiality
  - Availability
  - Non-Repudiation
  - …
- Mobility
- Agent Execution

Different Points of View

- Protection of Platforms
- Protection of Agents
Table of Contents

1 Foundations
   Agents and Multi Agent Systems
   Agents and Malware

2 Security in MAS
   Desirable Properties
   Protecting the Platform
   Protecting the Agent

3 Conclusion
Approaches to Protection

Briefly

- Reference Monitor
  - Security Kernel
  - Sandbox
- Signed Code
- Path Histories
- State Appraisal
- Proof Carrying Code
Approaches to Protection

Briefly

- Reference Monitor
  - Security Kernel
  - Sandbox
- Signed Code
- Path Histories
- State Appraisal
- Proof Carrying Code

⇒ Not the focus of this presentation
State Appraisal

Description

- Assurance to Platform that Agent will not reach certain states
- Appraisal functions become part of Agent’s code
- State Space Explosion
- Requires Prediction of all (harmful) States
Proof Carrying Code

Description

- Executor (e.g. Platform) can check Program/Code (e.g. Agent)
- Dynamic Approach
- Code comes with Proof not to violate Policy
- Generation of Proof difficult
- Validation of Proof easy
- Does not solely rely on States
Table of Contents

1 Foundations
   Agents and Multi Agent Systems
   Agents and Malware

2 Security in MAS
   Desirable Properties
   Protecting the Platform
   Protecting the Agent

3 Conclusion
Approaches to Protection

Overview

- Trusted Hardware
- Policies
- Logging
- Cooperation
- Cryptography
- Code Obfuscation
Trusted Hardware

Description

- Probably best Protection Possible
- Hardware can be tampered with, too
  - Power Supply, Voltage
  - Timing
  - Information Leaking
  - …

Trusted Computing

- Needs Trusted Hardware
- Other Issues (e.g. DRM)
Trusted Hardware

Description

• Probably best Protection Possible
• Hardware can be tampered with, too
  • Power Supply, Voltage
  • Timing
  • Information Leaking
  • …

Trusted Computing

• Needs Trusted Hardware
• Other Issues (e.g. DRM)

⇒ Not relevant for this analysis
Policies

Description

- Recommended for any Setup
- Regulatory Approach
- “Prohibit” Malicious Activity
- Enough for certain Scenarios

Problematic

- Enforcement of Policies
  - Prevention of Violations
  - Sanctions after Violations
- Employ together with Logging
Policies

Description

- Recommended for any Setup
- Regulatory Approach
- “Prohibit” Malicious Activity
- Enough for certain Scenarios

Problematic

- Enforcement of Policies
  - Prevention of Violations
  - Sanctions after Violations
- Employ together with Logging

⇒ Not relevant for Malware
Logging

Description

- Keep a History of Actions
- Possibly with Signatures
  - Platforms
  - Agents
- Useful in conjunction with Policies

Problematic

- Logging alone does not prevent most Incidents
- Sanctioning is supported
Logging

Description

- Keep a History of Actions
- Possibly with Signatures
  - Platforms
  - Agents
- Useful in conjunction with Policies

Problematic

- Logging alone does not prevent most Incidents
- Sanctioning is supported

⇒ Not relevant for Malware
Cooperation

Description

- Distribution of Information or Functionality
- Simply Redundancy
Cooperation

Description

- Distribution of Information or Functionality
- Simply Redundancy

⇒ Redundancy often at least implicitly present
Cryptography

Main Question

- Cryptography on Untrusted Platform

Overview

- Partial Results Encapsulation
- Computing with Encrypted Functions
- Undetachable Signatures
- Environmental Key Generation
- Secure Communication
Partial Results Encapsulation

- Secure Data Storage for Agent
- Several Approaches in Literature
- Encrypt Data with Public Key (e.g. owner’s)
- Useful for collecting data from several Platforms
- Agent cannot use Data
- Current Platform sees Data
- Signatures can be problematic
Partial Results Encapsulation

- Secure Data Storage for Agent
- Several Approaches in Literature
- Encrypt Data with Public Key (e.g. owner’s)
- Useful for collecting data from several Platforms
- Agent cannot use Data
- Current Platform sees Data
- Signatures can be problematic

⇒ Applicable to Malware
Cryptography (cont’d)

Computing with Encrypted Functions

- $f()$: Function to be run by Agent
- $enc()$: Function to encrypt (hide) Information from Platform
- $g = f \circ enc$: Function executed on Platform
- Platform knows: $g()$, might also know $enc()$
- Platform cannot compute $f(x)$, only $g(x) = enc(f(x))$
- $enc()$ not easy to find
- $f(x)$ might be needed by Agent
- Denial of Service, Replay Attacks
Cryptography (cont’d)

**Computing with Encrypted Functions**

- $f()$: Function to be run by Agent
- $enc()$: Function to encrypt (hide) Information from Platform
- $g = f \circ enc$: Function executed on Platform
- Platform knows: $g()$, might also know $enc()$
- Platform cannot compute $f(x)$, only $g(x) = enc(f(x))$
- $enc()$ not easy to find
- $f(x)$ might be needed by Agent
- Denial of Service, Replay Attacks

⇒ Applicable to Malware
Cryptography (cont’d)

Undetachable Signatures

- Application of Computing with Encrypted Functions
- \( f() \): Agent’s Signature Function
- \( enc() \): Also includes Agent’s Constraints
- \( x \): Contract to be signed
- \( g(x) = enc(f(x)) \): Agent’s Signature of Contract
- \( enc() \) restricts what can be signed

Till Dörges
Undetachable Signatures

- Application of Computing with Encrypted Functions
- $f()$: Agent’s Signature Function
- $enc()$: Also includes Agent’s Constraints
- $x$: Contract to be signed
- $g(x) = enc(f(x))$: Agent’s Signature of Contract
- $enc()$ restricts what can be signed

⇒ Applicable to Malware
Cryptography (cont’d)

Environmental Key Generation

- Unlock Code (or Data) based on Condition in the Environment
- Condition Encoded Using Hash Functions
- Code available in clear just before Execution
Environmental Key Generation

- Unlock Code (or Data) based on Condition in the Environment
- Condition Encoded Using Hash Functions
- Code available in clear just before Execution

⇒ Applicable to Malware
Cryptography (cont’d)

Secure Communication

• Securing Command and Control Channels inside Network
• Hiding Contents from Platform not possible
• Undetachable Signatures applicable
Secure Communication

- Securing Command and Control Channels inside Network
- Hiding Contents from Platform not possible
- Undetachable Signatures applicable

⇒ Applicable to Malware
Description

- Perfect Obfuscation $\equiv$ Perfect Information Hiding
- Obfuscation $\neq$ Encryption
- Perfect Obfuscation impossible
- Current Quality of Obfuscation
  - leaking of “negligibly small” amount of information
  - polynomial time
Code Obfuscation

Description

- Perfect Obfuscation $\equiv$ Perfect Information Hiding
- Obfuscation $\neq$ Encryption
- Perfect Obfuscation impossible
- Current Quality of Obfuscation
  - leaking of “negligibly small” amount of information
  - polynomial time

$\Rightarrow$ Applicable to Malware
# Table of Contents

1. Foundations
   - Agents and Multi Agent Systems
   - Agents and Malware

2. Security in MAS
   - Desirable Properties
   - Protecting the Platform
   - Protecting the Agent

3. Conclusion
Conclusion

Summing up

- Advanced Protection Possible for Malware
- Perfect Protection Impossible
- Some Measures Used already

Not to forget

- Turing and the Entscheidungsproblem
- Current Malware already “successful”
- Complexity of Current Setups makes for good Hiding Spots
Remains ...

- Thanks for your Attention!
Remains ...

- Thanks for your Attention!
- Questions?