

*Experimental Methods
in Software Engineering:
Anomaly Detection
WS 1999/2000*

Allen Dutoit

dutoit@in.tum.de

Technische Universität München

Institut für Informatik

Lehrstuhl für Angewandte Softwaretechnik (XII)

Preliminaries

- This seminar is conducted in English
 - ... but don't worry too much about it :^)
- This seminar counts as an:
 - “Überfachliches Grundlagenseminar für Informatiker”
- Lehrstuhl für Angewandte Softwaretechnik (XII)
<http://www12.in.tum.de/>
 - STARS Softwaretechnikpraktikum
 - HS Project Management (mailto:bruegge@in.tum.de)
 - Diplomarbeiten/SEP

Overview

- Goals
- What is experimentation?
- What is anomaly/intrusion detection?
- Seminar format
- Organization

Hauptseminar: goal

- Introduction to scientific work
- Learning by doing: each student learns
 - to investigate a specialized topic related to the seminar's theme,
 - to search relevant literature,
 - to select what is important, and
 - to present it succinctly to others.

Goals of this Hauptseminar

- Emphasize the importance of experimental methods in software engineering
- Provide an overview of anomaly detection as application domain
- Provide hands on experience in using experimental methods

What is Experimentation?

- Experimentation helps determine the effectiveness of
 - a theory (e.g, relativity)
 - a tool (e.g., C++, Java, CASE)
 - a method (e.g., object-oriented analysis & design)
- Experimentation requires the collection and analysis of data
- Experimentation can include:
 - Literature search
 - Case study
 - Synthetic experiment
 - Replicated experiment
 - Quasi experiment

*An example of **lack** of experimentation*

- A large segment of the industry converted from C to C++
- Decision was based on the benefits of OO
- Decision was **not** supported by experimental data
- Currently, there are experiments showing that C++ programmers make more mistakes, during both development **and** maintenance.

What is intrusion detection?

An intrusion can be defined as:

“Any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource.”

Internet worm 1988

- 3000-4000 computers were infected (about 5% of the internet)
- Many ghost processes were consuming CPU time
- Killing these processes did not seem to help
- Rebooting machines did not cure the problems
- The problem only occurred on sun's and vax'en

Internet worm overview

- Internet worm propagated by exploiting three different vulnerabilities:
 - sendmail debug mode
 - fingerd buffer overrun
 - accounts with no or weak passwords
- Several features were designed to conceal its identity
 - command shell was zero'ed out
 - all strings in the binary were XORed
- Once on the machine, the worm would collect information:
 - /etc/hosts
 - .rhost files

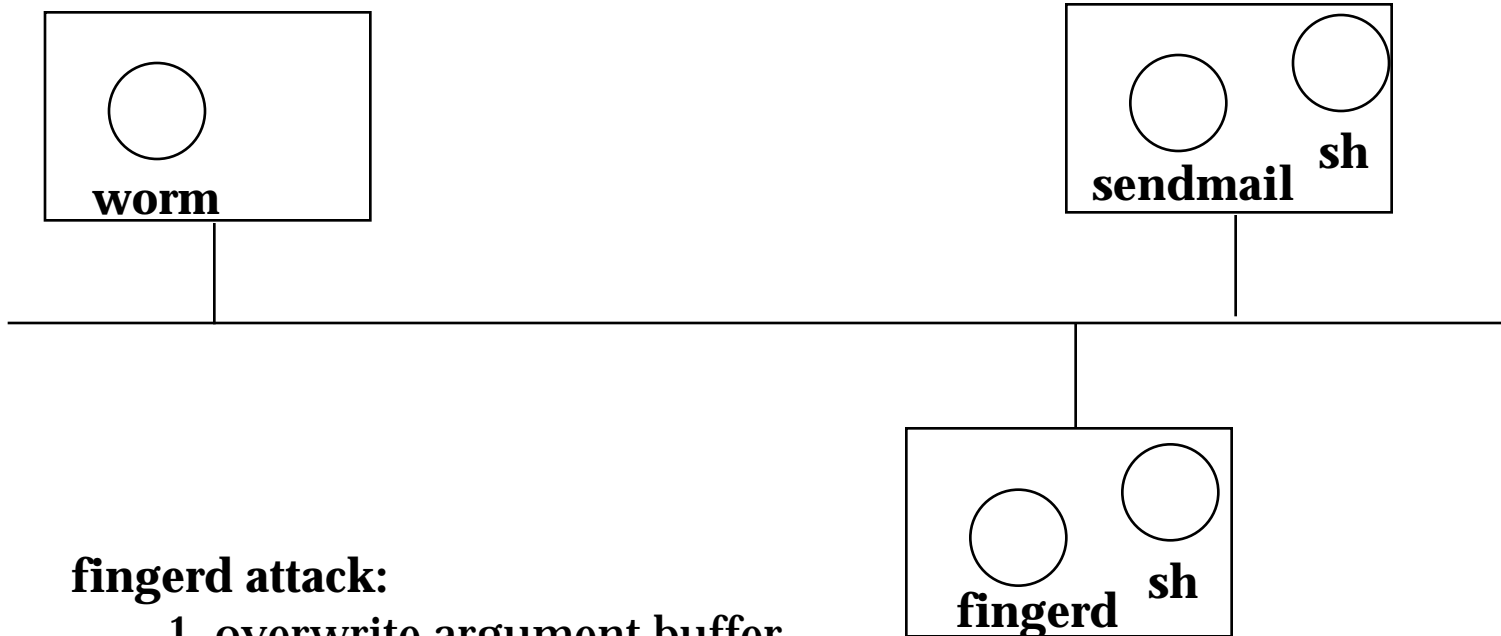
Internet worm: propagation

local attack

1. try passwords from a dictionary
2. use `rsh` to exploit network of trust

sendmail attack:

1. put `sendmail` in debug mode
2. have `sendmail` fork `sh`
3. use the shell to download and compile a new worm



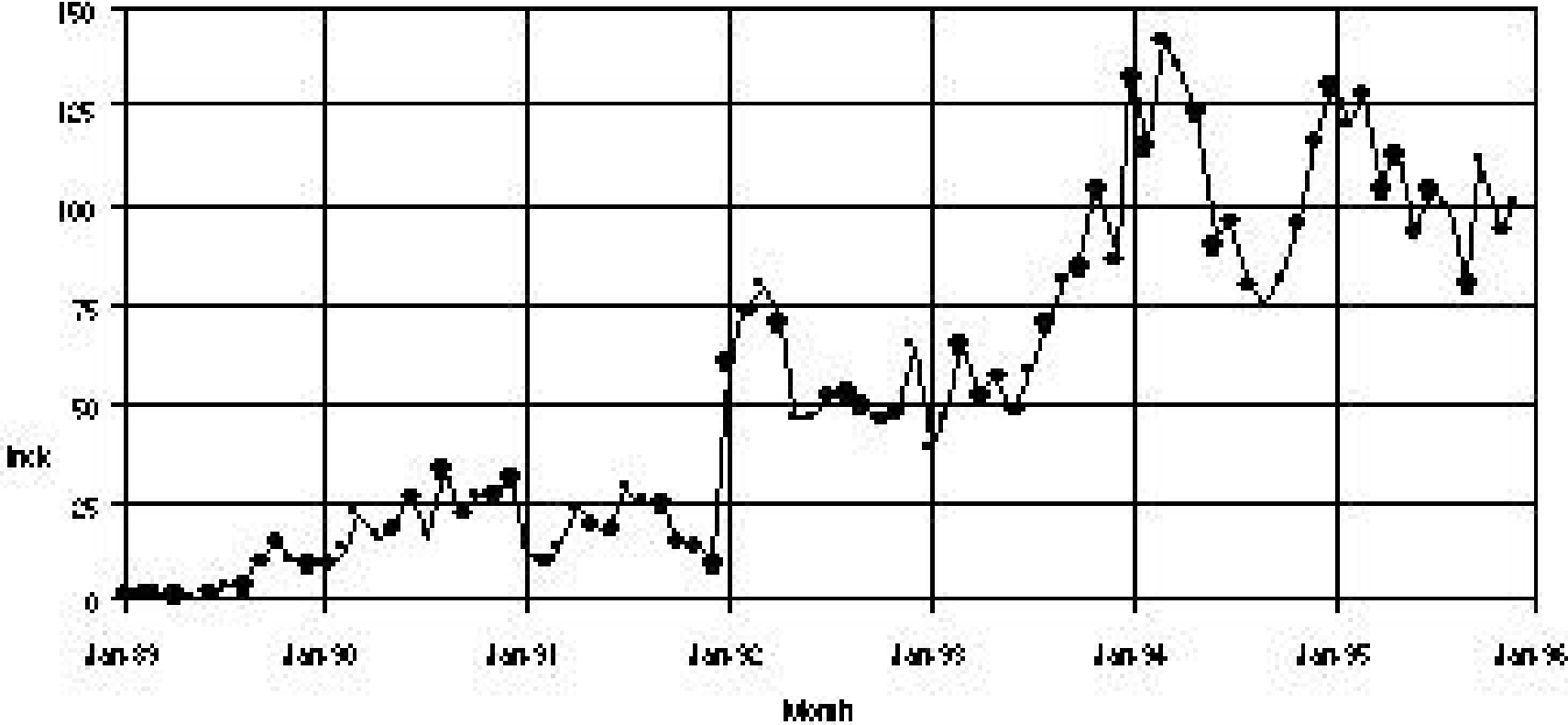
fingerd attack:

1. overwrite argument buffer and replace `finger` with `sh`
2. use the shell to download and compile a new worm

Internet worm: aftermath

- Estimated damage
 - 5% of the internet affected (80'000 nodes)
 - Disrupted e-mail, work at many universities and research institutions
 - Thousands of sysadmin hours
 - Possibly several millions of dollars in total costs.
 - The internet took 1 week to recover.
- Robert T. Morris was
 - suspended for 1 year from Cornell
 - convicted of 'Federal Computer Tampering'
 - \$10'000 of fine, 400 hours of community work, and 3 years probation
- CERT was created ...

CERT: Trends



Trends (cont'd)

- 1988: Internet Worm
- 1995: Kevin Mitnick
- 1996: 250 US DoD computers penetrated
- 1997: 500 were expected
- 1999: in one incident 250 unclassified computers of the Marines Corp. were corrupted by a virus. One day of downtime and many documents lost.

Intrusions: definitions

An intrusion can be defined as:

“Any set of actions that attempt to compromise the integrity, confidentiality or availability of a resource.”

Intrusions can be categorized into two main classes:

- ***Misuse intrusions***
 - well defined attacks on known weak points of a system
 - detected by watching for certain actions being performed on certain objects.
- ***Anomaly intrusions***
 - based on observations of deviations from normal system usage patterns.
 - detected by building up a profile of the system and detecting significant deviations from this profile.

Intrusion detection and experimental methods

- Intrusion detection systems, in particular anomaly detectors, are designed to detect present and **future** intrusions.
- In the presence of such underspecified requirements, anomaly detectors are imperfect:
 - They may detect anomalies that are not intrusions
 - They may fail to detect an intrusion
- Experimental methods, consequently, play a critical role in debugging and evaluating such systems.

Seminar format

- Introduction to experimentation and bibliographical research (Allen)
 - Why experiment?
 - How to experiment?
- Paper presentations (10 students)
 - What do other people do out there?
- Intrusion detection algorithm
 - Do your own experiment.
- Wrap up (Allen)

Paper presentations

- What is the paper about? (15 min.)
 - What is the problem domain?
 - What is the problem?
 - What is the approach taken by the researcher?
 - What are the results?
- What have others done? (15 min.)
- IYHO, what do *you* think? (15 min.)
 - What's wrong with the paper?
 - What's good about it?
 - What are the open issues?

More on this next week...

Topics

24.11.1999	<i>Security vulnerabilities in computer programs(Schraegle)</i>
1.12.1999	<i>Evaluation of Intrusion Detection Tools (Borgwardt)</i>
8.12.1999	<i>Network Security Monitor (Rost)</i>
15.12.1999	<i>Distributed Intrusion Detection (Koukouchkine)</i>
22.12.1999	<i>Pattern Matching (Löhr)</i>
12.1.2000	<i>Intrusion Detection Using Statistical Methods (Wilm)</i>
19.1.2000	Rule-Based Approach
26.1.2000	Specification-Based Approach
2.2.2000	Genetic algorithms for ID
9.2.2000	<i>Real-time intrusion detection (Stan)</i>

Intrusion detection algorithms

In addition to a presentation, you will develop a small intrusion detection algorithm and evaluate it experimentally.

Assumptions:

- The algorithm analyses the sequence of system calls in the system service being monitored
- The algorithm can be trained using an intrusion-free trace
- Intrusions will manifest as an anomaly

Intrusion detection algorithms (cont'd)

Evaluation framework:

- The algorithm will be trained using synthesized traces with increasing noise
- The algorithm will then be evaluated using synthetic traces with injected anomalies

Types of injected anomalies:

- Foreign symbol - sequence containing a symbol that never occurs in the training trace
- Foreign tuple - sequence that never occurs in the training trace
- Rare tuple - sequence that occurs less than 5% in the training trace

Intrusion detection algorithms (cont'd)

Examples

Regular string: "ABABABAAJABABABABABA"

Noisy string: "ASDAAJHFHSADDHJLASD"

Anomalies (n=3):

Foreign symbol: "ABX"

Foreign tuple: "AAA"

Rare tuple: "AAJ"

Intrusion detection algorithms (cont'd)

Infrastructure:

- A web site will be provided to automate the evaluation
- The algorithm should be written in C

More on this next week ...

Infrastructure

- Web site
 - <http://www12.in.tum.de/teaching/WS99/ese.html>
- Access to the library of the Institut für Informatik
- **Office hours**
 - Mondays 14-16 in -1207 or by appointment

Bibliography

- [Bellovin, 1992] Bellovin, Steven M., “There Be Dragons”, In UNIX Security Symposium . Baltimore, Maryland: USENIX Association, 14-16 September 1992, pp. 1-16.
- [Bishop et al., 1997] Bishop, Matt; Cheung, Steven and Wee, Chris, “The Threat from the Net”, IEEE Spectrum, Vol. 34, No. 8, August 1997, pp. 56-63.
- [Forrest et al, 1997] Forrest, Stephanie; Hofmeyr, Steven A. and Somayaji, Anil, “Computer Immunology”, Communications of the ACM, Vol. 40, No. 10, October 1997, pp. 88-96.
- [COAST, 1999] Intrusion detection bibliography and introduction
<http://www.cerias.purdue.edu/coast/intrusion-detection/introduction.html>