

March 31th, 2004

Stefan Lampart (Open Systems AG); Thomas Dübendorfer, Arno Wagner, ETH Zürich

Master's Thesis:

Scan Detection Based Identification of Worm-Infected Hosts

for Roman Hiestand <romhiest@ee.ethz.ch> and
Christoph Göldi <goeldich@ee.ethz.ch>

1 Introduction

Computer viruses in general

In recent years a lot of computer worms and viruses have been spreading rapidly all over the world (e.g. SQL Slammer, MSBlaster, Sobig, Welchia worm). These computer worms can be destructive in multiple ways:

- Information may be stolen or deleted from infected hosts.
- An infected host may try to infect other hosts and generates lots of traffic, resulting in overloaded networks, VPN tunnels and gateway machines.

It is predicted that in the next years the number of new worms and viruses will increase rapidly.

Problems of pattern-based virus detection

There are already tools that allow to search for worms and viruses in the Internet traffic (e.g. SMTP or HTTP) or in files on servers or hosts. These tools are known as virus scanners and are looking for known patterns of malicious code. The drawback of these systems is that they cannot detect unknown worms and that they have to be regularly updated with the latest virus recognition patterns.

Scanning by worm-infected hosts

As soon as a worm has infected a host it tries to scan others and then to infect the vulnerable hosts. Some worms try to propagate in e-mails that are sent to every known e-mail account,

other worms try to spread by directly scanning for vulnerable machines in the local network or in the Internet. They try to connect to thousands of hosts in a certain range with a certain protocol. As soon as they find a host that answers to their request they try to infect it.

Scanning detection

The process of looking for hosts that are answering on certain ports is known as scanning. As during such scans characteristic traffic is generated such scanning activity can be detected. This methodology is known as scanning detection. The main idea of this thesis is to detect scans originating from worm-infected hosts in a private LAN by monitoring network traffic between the private LAN and an external network (such as the Internet or another private LAN connected through VPN Gateways).

2 The Task

The thesis is conducted at Open Systems AG¹ in Zürich. The task of the students is split into four major subtasks that all will be: analysis of known scanning mechanisms of worms, analysis of existing scanning detection algorithms, specification of generic infected-host detection techniques using scanning detection and implementation of a prototype.

Analysis of known scanning mechanisms of worms

The scanning mechanisms of known worms have to be studied and analyzed. The different worms should be classified regarding their scanning mechanism. It then should be possible to decide which classes of scanning mechanisms can be discovered by using scanning detection. For each class a tool should be implemented that generates network traffic similar to a worm of this class.

Analysis of existing scanning detection algorithms

Existing scanning detection mechanisms and algorithms should be investigated and analysed. The most promising algorithms should be implemented and tested against the implemented worm simulation tools. The results of these tests have to be compared.

Specification of generic infected-host detection techniques using scanning detection

The students should find out which scanning detection algorithms can be used to detect which class of worm scanning mechanisms. They also should try to propose new algorithms that can be used to detect scanning worms. They should try to find a generic algorithm that also can detect new worms. A specification for such a worm scan detector has to be written.

¹See <http://www.open.ch/>

Implementation of a prototype

Following the above mentioned specification a prototype should be implemented on Linux (and run possibly also on Solaris). It should run on a (Linux or Solaris based) VPN gateway computer that connects an internal LAN with an external network. The resource consumption and performance of the prototype must be such that the gateway can still offer its normal services (e.g. VPN, NAT etc.). After some traffic analysis the prototype should output a list of possibly worm-infected hosts together with some meaningful parameters that indicate the reliability of the detection per host and that explain the type of worm-infection detected.

Testing and test setup

The prototype must be thoroughly tested with real traffic and under real network load. Therefore a small testbed (at least four computers are needed: one for LAN 1, two for the two VPN gateways and one for LAN 2) should be provided by Open Systems. In addition, a capture of real productive network traffic as seen by a VPN gateway must be organized. The tools created for generating worm scanning traffic should be used to inject traffic during a replay of the capture and in a later phase to directly generate worm scanning traffic in a larger test setup. At a later stage of the prototype development, the “Scylla”-cluster² at ETH (up to 23 machines at 1 Gbit/s speed) or the TIK computer lab room (up to 18 machines and 3 routers at 100 Mbit/s) could be used during some days for a bigger test setup and for observing traffic at higher speeds.

By analysing the results of these tests the detection parameters of the prototype should be tuned and the code improved. With the help of the prototype statistics about the efficiency of the implemented scan detection algorithms should be made. Clear statements about how to fine-tune the parameters of the scan detection algorithms for reducing false positives and false negatives are expected after the tests.

3 Deliverables

The following results are expected:

1. *Survey of scanning mechanisms of known worms* A short but precise survey that studies and analyzes the different worm scanning mechanisms. This survey should also classify the different mechanisms.
2. *Scanning mechanism simulation tools* Tools that generate network traffic of at least two different classes of scanning mechanisms have to be implemented.
3. *Scanning detection survey* A survey that summarizes the scanning detection algorithms already known in the literature.
4. *Implementation of scanning detection algorithms* The studied scanning detection algorithms should be implemented and validated by feeding them with the spreading mechanism tools.

²See <http://www.tik.ee.ethz.ch/~ddosvax/cluster/>

5. *Specification of a prototype* In this creative phase the students should find and write down a specification of a generic algorithm. This generic algorithm should be capable of detecting new unknown worms.
6. *Implementation of a prototype* The specified prototype should be implemented.
7. *Testing and tuning of the prototype* Tests of the prototype with real traffic should be made in order to validate the functionality. The prototype must support the detection of at least two different scanning mechanisms.
8. *Documentation* A concise description of the work conducted in this thesis (task, related work, environment, code functionality, results and outlook). The two surveys as well as the descriptions of the prototype and the testing results are part of this main documentation.

Further optional components are:

- Implementation of other scanning or spreading detection algorithms e.g. detection of mail worms.
- Graphical plots of the statistics extracted from the tests of the different scanning detection algorithms.
- Real time graphical plots of worm scanning activity detected by the prototype.
- Paper that summarizes in ten pages the task and results of this thesis.

Documentation and presentation

A documentation that states the steps conducted, lessons learnt, major results and an outlook on future work and unsolved problems has to be written. The code should be documented well enough such that it can be extended by another developer within reasonable time. At the end of the thesis, a presentation will have to be given at TIK that states the core tasks and results of this thesis. If important new research results are found, a paper might be written as an extract of the thesis and submitted to a computer network and security conference.

The developed code of the prototype and the implemented algorithms will be released under the terms of GPL2 as open source at the end of the thesis.

Dates

This Master's thesis starts on Monday, October 18th, 2004 and is finished on Monday, April 18th, 2005. It lasts 26 weeks in total.

Two intermediate informal presentations for Prof. Plattner and all supervisors will be scheduled 2 months and 4 months into the thesis.

A final presentation at TIK will be scheduled close to the completion date of the thesis.

Informal meetings with the supervisors will be announced and organized on demand.

Supervisors

Stefan Lampart, stl@open.ch, +41 1 455 74 00, Open Systems AG, <http://www.open.ch>

Thomas Dübendorfer, duebendorfer@tik.ee.ethz.ch, +41 1 632 71 96, ETZ G64.1

Arno Wagner, wagner@tik.ee.ethz.ch, +41 1 632 70 04, ETZ G64.1