Subject

Distributed Denial of Service (DDoS) are a threat to Internet services ever since the widely published attacks on e-bay.com and amazon.com in 2000. ETH itself was the target of such an attack 6 months before these commercial sites where hit. ETH suffered repeated complete loss of Internet connectivity ranging from minutes to hours in duration. Massively distributed DDoS attacks have the potential to cause major disruption of Internet functionality up to and including severely decreasing backbone availability.

Attack Model

Most DDoS attacks share a common pattern: An infection phase where the initiator acquires the attack resources by compromising a large number of weakly protected hosts, ideally causing little or no visible change in host behavior, in order to make the compromise hard to notice. An infection phase can last from less then 10 minutes to several months. Attacks within the order of 100,000 and more compromised hosts have already been observed in practice (Code Red, Sapphire).

In a second phase, the attack phase, the attacker uses the compromised hosts to initiate actual attacks on a target computer or network. These attacks can be done autonomously or under direct or indirect control of the attacker. Although attack control increases the risk of identification for an attacker, there are possibilities to keep this risk small.

Semester Thesis Task

The focus of this semester thesis is to use simulation to explore the parameter space for worms. Typical parameters are time to infect, time to scan a host, number of parallel scan probes,
scanning strategy, amount of data needed for infection, and others. While some worms may scan and infect at the same time with a single UDP packet, others need an explicit host and vulnerability scan first via TCP and then a second TCP connection to actually infect the target.

Different choices for these parameters lead to different spreading speeds, different patterns in the observed scanning and infection traffic and different side-effects, like network overload.

The tasks is now to identify ideal choices for these parameters, together with the resulting observable traffic characteristics, as well as a partial exploration of sub-optimal parameter choices. The main goal is to create a method that allows deduction of a rough approximation of the worm parameters from observed traffic patterns.

This task is split into the following subtasks:

**Understand the possible methods a worm can use to propagate and identify the associated parameters and traffic patterns resulting from them**

Worms can use different protocols and mechanisms to propagate themselves. This semester thesis should mainly be concerned with "online mechanisms" where no human interaction is involved. Email worms, e.g., are only an optional subject of this semester thesis.

The approach used could be an in-depth look at past worms and building on this a theoretical (short) exploration of other possible methods.

**Selection of the Internet model(s)**

This simulation should be a quantitative simulation that abstracts from individual hosts and instead regards groups of hosts that have a certain type of interconnectivity and a certain type of connectivity to the rest of the Internet. While a precise model of the Internet is possibly infeasible, some groups of hosts can be identified, e.g. home users with dial-up connections, users with (A)DSL/Cable modem, hosts that are always connected to the Internet with medium to high bandwidth. These groups can be further divided into geographical groups.

If the available literature does not yield a suitable model, the candidate should come up with simple approximate models for today’s Internet. To simplify this task, subnet structure and characteristics should be modeled only roughly, so that the total number of host groups identified does not exceed 100-1000 groups.

If this is insufficient to simulate propagation strategies that treat local subnets differently from the rest of the Internet, this semester thesis should only try to refine the model if there is enough time left.

**Design and Implement a simulator**

The student will design a simulation architecture that is appropriate for the Internet model(s) chosen in the previous step. Care will be taken that the architecture is modular enough to allow its partial or full re-use in future work.
After that, the student will implement a prototypical simulator based on this architecture using the scripting language Perl. If some components turn out to be especially time-critical, these components will be replaced by optimized implementations in C, as far as time permits.

Graphical analysis is not required and will be implemented only when time is left for this task.

**Do simulation to explore the parameter space**

The created simulator should allow simulation of worm spreading patterns for worms with different characteristics with regard to

- Scanning strategy
- Time and network traffic needed to identify a vulnerable target
- Time and network traffic needed to infect a target
- Number of parallel scans and infections that can be in progress
- Bandwidth-limited or latency-limited spreading speed and possible trip-points where latency-limited worms turn into bandwidth-limited ones
- Number of vulnerable hosts

and possible other parameters. Parameters will be identified both by analysis of known worms as well as known remote exploits. Worms that spread very slow (e.g. more than a day to 90% infection) are not in the focus of this work.

The main characteristics that should be the result of a simulation is a time vs. infection percentage statistic and statistics of the main parameters of the generated scanning and infection traffic (amount vs. time).

The student will identify "reasonable" ranges for the major parameters. Based on this parameter space interesting areas, like extremely effective strategies, ineffective strategies and strategies with surprising strategies will be identified with simulation. The objective is not a full exploration of the parameter space, but rather a very rough map that identifies interesting areas.

**Compare results with worms observed in the past**

In order to validate the simulation model and to estimate the accuracy of the simulation results, some simulations will be done with parameters from past worms and the results will be compared with the observed behavior of those worms. This can also help to identify critical parameters and to understand known worms better.
Documentation and Presentation

A documentation that states the steps conducted, lessons learned, 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 programmer within reasonable time. At the end of the semester, a presentation will have to be given at TIK that states the core tasks and results of this semester 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/or security conference.

Dates

The semester thesis will start on March 31, 2003 and should be finished by end of May 2003.

Time-line

Numbers are weeks from the first week, for a total of seven weeks full time work on the semester thesis.

1. Literature search, overall understanding of concepts.
2. Worm propagation, definition of the Internet model.
3. Design and implement a simulator, first simulations at end of the week
4. Improve the simulator and start more extensive simulation. Possible adjustments to the Internet model when specific weaknesses turn up.
5. Simulation and results gathering. Probably some changes in the simulation tool if needed, such as optimizations.
6. Comparison to worms observed in the past.
7. Documentation and presentation.

Supervisors

Arno Wagner, wagner@tik.ee.ethz.ch, +1 632 7004, ETZ G64.1
Thomas Dübendorfer, duebendorfer@tik.ee.ethz.ch, +1 632 7196, ETZ G64.1