Objectives and Motivations

Objective:
- Improve modeling of propagation dynamics in botnets

Motivations (and limits of state-of-the-art models):
- Rank botnets and prioritize responses against them
- Model short-term variations in botnet populations
- Recognize that not all vulnerable populations are the same

Contributions
- Alternative method to collect control traffic of a botnet
- Novel botnet propagation model that takes into account time zones
  - Specific worm propagation studies
  - Models for specific propagation methods (email, hit list, etc.)
  - Models in presence of specific defense mechanisms

Building a Dataset

Assumption:
- Botnet has already been detected (through honeynet, tarpit, etc.)
  2 steps:
  - Perform malware analysis to identify resolution of C&C server’s name
    - Assumption: centralized architecture
    - Assumption: DNS-based name resolution
  - Contact registrar and create a sinkhole for the C&C server’s name (legitimate DNS poisoning)
    - Effect: all traffic from bots to C&C redirected to sinkhole and collected

Methodology

Sinkhole implements several mechanisms to “improve” dataset:
- Collects traffic only for non-public servers (no extra legitimate traffic)
- Completes 3-way TCP handshake (no random scans)
- Sets TCP window size to 0 (reduced impact of DHCP churn)

Effects:
- Capable of distinguishing different botnets
- Map victims to respective botnet
Results

- 5 months of data collection
- 50 botnets
- Different size characteristics (as noted in past weeks papers)

Insight

- Strong diurnal nature of botnets
- Key parameters of precise model:
  - Time of day
  - Region of spreading

Time Zone-based Propagation Model

- SIR (Susceptible/Infected/Recovered) epidemiological model
- Extension of the classic Kermack-McKendrick model

Definitions

- Diurnal shaping function $\alpha(t)$: fraction of computers with the vulnerability exploited by a botnet that are on-line
- Infection rate $\beta = \eta/\Omega$ (scanning rate over IP space scanned)
- $N(t)$: initial number of vulnerable hosts
- $I(t)$: number of infected hosts
- $S(t)$: number of hosts that may be infected
- $R(t)$: number of recovered hosts (patched, down, etc.)

Models

Single time zone model:

$$\frac{dI(t)}{dt} = \beta P(t)S(t) - \frac{dR(t)}{dt} =$$

$$= \beta \alpha^2(t)f(t)[N(t) - I(t) - R(t)] - \gamma \alpha(t)I(t)$$

K time zones model:

$$\frac{dI_i(t)}{dt} = \sum_{j=1}^{K} \beta_j I_j(t)S_j(t) - \frac{dR_i(t)}{dt} =$$

$$= \alpha_i(t)[N_i(t) - I_i(t) - R_i(t)] \sum_{j=1}^{K} \beta_j \alpha_j(t)I_j(t) - \gamma \alpha_i(t)I_i(t)$$

Parameters

- 3 time zones grouping North America, Europe and Asia ($K = 3$)
- $I_i(t)$ proportional to measured number of incoming SYN packets
- $\beta_j$ uniform
- $N_i(t)$ constant (max $I_i(t)$ ?)
- $\alpha_i(t)$: approximated with polynomial that minimizes the cumulative square error with respect to average measured $I_i(t)$ on 24-hour segments (note that this really measures $I_i(t)$ and not the number of vulnerable hosts over time: assume proportional?)
- $R(t)$? Assume 0?

A numerical solution for $I_i(t)$ is derived using Matlab
Validation

Validation on the European group:

![Graph showing botnet data, SIR model, and Churn model over time.]

Forecasting with Pattern Tables

Key fact:
- $\alpha(t)$ is both vulnerability and time zone-specific

Claim:
- Can be used to forecast the spreading of similar infections

Validation:
- Analyze spreading of 3 similar botnets (same vulnerability, therefore same expected $\alpha(t)$)
- Parameters are determined for botnet 1
- Propagation dynamic is normalized and curve-fitted on botnet 2, 3
- Verification of expected behavior on botnet 2 and 3

Optimal Release Time

- What is the influence of the release time on the propagation of an infection?
- Simulation on Code Red-like worm
  - $N_1 + N_2 + N_3 = 360K$
  - $\eta = 358/min$

Limitations

- Model predicts dynamics, not infected populations
- Value of derived $\alpha_i(t)$ decays with time (users upgrade, machines are replaced, patches applied, etc.): it can be applied only for a limited amount of time
- $\alpha_i(t)$ is vulnerability-specific (as opposed to botnet-specific): needs to be refined to deal with botnets that target multiple vulnerabilities
- Method does not matter for very slow spreading and very fast spreading worms, but only for middle spreading (Code Red: 14 hours)

Questions?

- How can the model be adapted to real-world scenarios?
- What are the limitations of using $\alpha(t)$ for forecasting botnet propagation?
- How can the model handle situations where multiple vulnerabilities are targeted by a botnet?