**T-DOMINANCE AND THE DISTRIBUTED ALGORITHM**

§ Reachability

From the connectivity logs [2] of a pair of encountered smartphones, we can estimate their reachability, i.e., the average interval between consecutive encounters. Given a set of smartphones \( P \), let \( G(P) \) be the undirected weighted graph with \( P \) as vertices and reachability as the weight on the edges; \( G'(P) \) be the subgraph of \( G(P) \) with weight-greater-than-\( T \) edges removed.

**Definition 1** (T-dominance). Let \( P \) be a set of smartphones and \( B \) be the set of bots in \( P \). \( B \) is a botnet which T-dominates \( P \) at time \( t \) if \( \forall B \in G'(P); \exists \forall p \in G'(P) \), either \( p \in B \) or \( p \) is a neighbor of a bot \( b \in B \) in \( G'(P) \).

§ Intelligence Exchange

Two kinds of encounters.

Nodes exchange intelligence at encounters. Two alternative approaches (raw and processed) are proposed.

**Prune and Infect**

Inspired by the Connected Dominating Set problem [3], we propose the prune-and-infect distributed algorithm for maintaining the T-dominance structural property.

**Prune** When a bot \( u \) meets another bot \( v \), \( u \) decides whether to disinfect (prune) itself (for stealthiness). We propose two alternative prune algorithms (individual and strong) based on two alternative priority-comparison criteria (strong and count).

**Infect** When a bot \( u \) meets a clean node \( v \), \( u \) decides whether to infect \( v \). The insight is that \( v \) should be infected unless it is likely to be pruned later. To decide the likelihood of \( v \) being pruned later, we check two criteria (prunable and coverage) consecutively.

§ Algorithm Properties

The prune-and-infect algorithm is localized and delay-tolerant in the following sense: if bot \( b \) prunes itself in its local (and potentially outdated) view at time \( t \), then, in the global (and updated) view, each of the smartphones \( T \)-dominated by \( b \), including \( b \) itself, is still \( T \)-dominated by some bot at \( t \) in the global view.

Due to the speculative nature of the reachability metric, the \( T \)-dominance structural property provides no hard guarantee that a non-bot will be reached by \( T \) after the attack time even if it is \( T \)-dominated by the botnet. However, our experiment shows that \( T \)-dominance provides a fairly good guarantee for reaching a majority in the smartphone pool.

**REFERENCES**


**RESULTS**

We use the dataset from the Wireless Topology Discovery (WTD) project [4] in our simulation. Botnet’s lifetime consists of two consecutive phases, herding and attack, with different goals. The goal of the herding phase is stealthiness as characterized by \( T \)-dominance; the goal of the attack phase is epidemic manifested in wide infection within \( T \).

Different prune strategies under \( T = 18,000 \) (5 hrs).

![Graph showing results](http://sysnet.ucsd.edu/wtd/data_download/wtd_data_release.tgz)

**CONTRIBUTIONS**

1. We propose the concept of botnet-level stealthiness and a novel structural property, \( T \)-dominance, for a stealthy botnet. Instead of infecting all susceptible smartphones, a stealthy botnet malware with the \( T \)-dominance property only infects those smartphones which can reach other smartphones within a time constraint of \( T \) with a high probability.

2. We design a distributed algorithm which maintains the \( T \)-dominance structural property and prove that the algorithm is localized and delay-tolerant in the sense that the algorithm maintains the structural property despite relying solely on local and potentially outdated information.

**THOUGHTS ON DEFENSES**

Cooperative defense. Proximity malware propagation circumvents centralized defense; individual nodes lack the resource to defend against malware. A cooperative defense mechanism which autonomously coordinates nodes on the task of malware defense is the way out.

Strategic sampling. A socially-aware malware sampling based on the \( T \)-dominance property will choose a socially well-connected group for malware detection. Prioritized patching. The \( T \)-dominance propagation can be applied in distributing malware patches. Instead of stealthiness, the balance between resource consumption and patch distribution efficiency are major concerns.