Honey Onions: Exposing Snooping Tor HSDir Relays

Guevara Noubir & Amirali Sanatinia
{noubir, amirali}@ccs.neu.edu
Northeastern University
Motivations

• Previous research studied the maliciousness of the relays
• Known bad Exit nodes
• Other work looked at the nature of hidden services content
• No prior work on the Hidden Service Directories (HSDirs)
• Indexing hidden services requires modification to Tor, which can be an indicator of some effort and potentially more malicious activities
Tor & Hidden Services

- A widely used practical anonymity infrastructure
- Provides anonymity for both the clients and the server through hidden services
- Depends on the honest behavior of the volunteering relays
- It is known that some relays are misbehaving (Bad Exit nodes)
- Some Exit nodes actively try to perform Man in the Middle Attack (MITM)
- Not much is known about the HSDirs or Hidden Services in general
Hidden Service Directories (HSDir)
Ring of Responsible HSDirs
Honey Onions (HOnions)

- Each HOnion corresponds to a server/process
- Run on local IP address (Hidden Service)
- Accessible only through Tor and not shared anywhere
- Three schedules
  - Daily
  - Weekly
  - Monthly
- Log the requests for further investigation
HOnions Architecture

1. Generate honions

2. Place honions on HSDirs

3. Build bipartite graph
Set Cover Problem

- \( HSD = \{ d_i : Tor \ relays \ with \ HSDir \ flag \} \)
- \( HO = \{ ho_j : HOnion \ that \ was \ visited \} \)
- \( V = \{ HSD \cup HO \} \)
- \( E = \{(ho_j, d_j) \in HO \times HSD \mid ho_j \ was \ placed \ on \ d_i \ and \ was \ visited\} \)
- \( \arg\min_{S \subseteq HSD} | S : \forall (ho_j, d_i) \in E, \exists d'_i \in S \land (ho_j, d'_i) \in E| \)

- The set cover is an NP-complete problem
- Can be calculated using approximation algorithms
- Set cover gives the lower bound on the number of snooping HSDirs
Heuristic Approach

• **Input:** $G(V, E)$: Bipartite graph of HOnions to HSDirs
• **Output:** $S$: Set explaining visits

$S \leftarrow \emptyset$

**while** $V \cap HO \neq \emptyset$ **do**
  • Pick $d \in V \cap HSD$: *with highest degree*
  • $V \leftarrow V \setminus \{d \text{ and its HOnion neighbors}\}$
• **end**
Integer Linear Programming (ILP)

- min \( (x_1, \ldots, x_{HSD}) \)
  \[ \sum_{j=1}^{\text{|HSD|}} x_j \]
  subject to \( \forall h_0 i \in HO \)
  \[ \sum_{\forall j: (h_0 i, d_j) \in E} x_j \geq 1 \]

- Provides a lower bound on the number of snooping HSDirs to explain the visits
Snooping Behavior

• Wide variety of behavior
• Automated vs manual probing
• Aggressive, periodic probing
• Attempts to find vulnerabilities
  • SQL Injection
  • XSS
  • Path traversal
  • PHP Easter Eggs
  • Targeting Drupal and Ruby on Rails
Snoopers’ Most Likely Geolocation
Snoopers’ Identity

• Hard to identify the real entity behind the relays
• More than half of the HSDirs are hosted on cloud platform
• The geolocations correspond to the location of the hosting platform and not necessarily the entity running them
• Number of cloud platforms are located in countries with stronger privacy protection for customers
• Some cloud platform accept payments over bitcoin, making it even harder to identify the real actors
Conclusion

• Honey Onions (HOnions) is a framework to detect snooping HSDirs
• Provides a lower bound on such relays
• Tor relies on the honest behavior of the volunteering relays
• The detection, identification and mitigation of misbehaving relays helps to improve the privacy and security of Tor
• This work is an addition to the previous body of work focusing on detection of misbehaving Tor relays