Requirements for Plan Recognition in Network Security Systems

Christopher W. Geib and Robert P. Goldman

Honeywell Labs
3660 Technology Drive
Minneapolis, MN 55418 USA
{geib,goldman}@htc.honeywell.com

Abstract. Computer network security systems need to incorporate artificial intelligence methods for plan recognition. Plan recognition is critical both to predicting the future actions of attackers and planning appropriate responses to their actions. This paper presents an argument for including plan recognition in such systems and discusses the problems network computer security presents to plan recognition research.

1 Introduction

For computer network security systems to move forward, they must be able to analyze the actions of a hacker[1], infer their goals, and make predictions about their future actions. In the artificial intelligence literature the deducing an agent's goals from observed actions is called intent or plan recognition or task tracking.

Other work in network security has argued for network level coordination among intrusion detection system (IDS) and even inferring attacker intent [1, 3, 7]. However, these papers have focused on the protocols and communication issues surrounding distributed coordination of information. This paper will argue for the explicit use of plan recognition, and highlight some of the required properties for such a system.

2 The Need for Plan Recognition

To be proactive, computer security systems must be able to infer the goals of attackers. To see this, consider the case of an IDS report of a synflood. In this case we assume that the attacker is using this synflood for one of two reasons.

1. a denial of service(DOS) attack,
2. Suppressing a host during an IP spoofing attack.

[1] We apologize for the use of the term "hacker" in its criminal sense, but we will use this as a convenient short hand in this paper

To correctly respond to this attack a network security system needs to understand the intent of the attacker, predict their next actions, and take steps to prevent the next actions. For example, in the DOS case, we can respond by modifying a firewall to reject packets from the attacking host or limit the number of connections to the DOSed machine. However to respond to an IP spoofing attack we should modify the firewall to prevent *all* external connections to all machines that trust the synflooded machine.

Notice that while the initial reported action is the same (synflood), the correct response is completely different. In fact the response in the first case will have no effect if the attacker's goal is access to another machine. By the time the firewall has been modified and the synflood clears the IP spoofing attack will have been executed and the attacker will likely already have access to the machine. Conversely if the attackers real intent is just to DOS the selected host, responding as though an IP spoofing attack is underway will cut off connections to other machines from the internet. In short, inferring the attacker's goal is critical to making predictions about the attacker's next action and thereby taking the correct countermeasures.

3 Requirements on Plan Recognition

The computer network security domain places a number of requirements on any plan recognition system that we would use. The following is a list of critical requirements for successful plan recognition in the network security domain.

Abandoning plans: Hackers regularly abandon goals and methods of achieving their goals. For example, if one line of attack proves fruitless they attempt a different exploit or if sufficiently frustrated in efforts to gain entry, they may choose to choose to DOS the machine instead.

Explicitly recognizing when a plan has been abandoned is a critical need in this domain. System's that don't do this will build up an ever increasing set of active or *open plans* that the agent has no intention of completing. A system attempting to find completions for these open plans will wind up considering unreasonable situations such as a hacker attempting the first step of an attack with a millisecond runtime but not attempting the second step of the attack until days or weeks later.

Hostile agents: Most previous work in plan recognition has assumed agents that do not mind or are even helpful in having their plans recognized. Unfortunately hackers are anything but cooperative in this regard. One of the first things an experienced hacker will do is to turn off system logging or at least delete the system logs so that their actions cannot be observed. Thus, in this domain a plan recognition system must be able to infer the execution of *unobserved actions* on the basis of other observed actions and observations of unexplained state changes.

Failure to observe: Suppose we observe an IPSweep. The longer we go without seeing any more hostile activity the more likely we are to believe that this was just an isolated scanning event. It was not part of a larger plan. However,

if right after the scanning event, we see other malicious activity then we are more likely to believe the scan is the reconnaissance step of a plan.

Our commitment to considering agents with multiple concurrent goals makes it even more critical that an assistant system be able to engage in this kind of reasoning. It is rare that we will be provided with definitive evidence that a hacker is *not* pursing a specific goal. Far more likely is that a lack of evidence for the goal over time will lower its probability.

Observations of failed actions: Current IDSs often report attacks that are not successful but have been attempted by the hacker. Since these attacks certainly indicate something about the attackers intent systems in this domain must take this kind of information seriously.

Partially ordered plans: The plans followed by attackers are often very flexible in the ordering of their plans steps. Consider the number of possible interleavings of IPsweeping methods and port sweeping methods that all provide the hacker with the same information: active IPs and open port numbers.

Multiple concurrent goals: Hackers often have multiple goals. For example, while the primary goal might be to use your computing resources to launch attacks against others the hacker might also be interested in stealing sensitive corporate data. Any successful system in this domain must be consider explanations that involve multiple goals.

Actions used for multiple effects: Consider scanning a subdomain. This action enables multiple actions including denial of service attacks as well as identifying web servers that might be defaced. This single action can be part of multiple plans. That is it can be *overloaded* for use by multiple plans.

Impact of world state on adopted plans: In many cases the activity of hackers is opportunistic. Finding a specific kind of operating system or running service may cause the hacker to try a specific attack first.

Ranking multiple possible hypotheses: Consider the case where all we observe is scanning activity. While this indicates a hacker is interested in our network, by itself it provides very little evidence about the hacker's intent. Rather than giving just one of the many equally likely answers, it is much more helpful to report the relative likelihood of each of the possibilities.

4 Conclusions

Many of these issues presented above have been explored in existing plan recognition literature [2, 4, 5, 6, 9, 8]. However, to the best of our knowledge, no research published to date covers all of these criteria. The first three of these criteria being the least studied. Thus, while the computer security domain can and should incorporate the existing work in plan recognition, the plan recognition community would do well to take up the computer network security domain on as a challenge problem to push the boundaries of the state of the art.

Acknowledgments

This material is based upon work supported by DARPA/ITO and the Air Force Research Laboratory under Contract No. F30602-99-C-0077.

References

[1] Intrusion detection message exchange format. `http://search.ietf.org/-internet-drafts/draft-ietf-idwg-idmef-xml-01.txt`.

[2] C. Conati, A. S. Gertner, K. VanLehn, and M. J. Druzdzel. On-line student modeling for coached problem solving using bayesian networks. In *Proceedings of the Sixth International Conference on User Modeling*, 1997.

[3] R. Feiertag, C. Kahn, P. Porras, S. Schnackenberg, S. Staniford, and B. Tung. A common intrusion detection language (CISL). Available at: `http://www.gidos.org/drafts/language.txt`.

[4] C. W. Geib and R. P. Goldman. Probabilistic plan recognition for hostile agents. In *Proceedings of the FLAIRS 2001 Conference*, 2001.

[5] R. P. Goldman, C. W. Geib, and C. A. Miller. A new model of plan recognition. In *Proceedings of the 1999 Conference on Uncertainty in Artificial Intelligence*, 1999.

[6] E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and K. Rommelse. The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, 1998.

[7] M.-Y. Huang and T. M. Wicks. A large-sace distributed intrusion detection framework based on attack strategy analysis. In *Recent Advances in Intrusion Detection (RAID98)*, 1998.

[8] H. Kautz and J. F. Allen. Generalized plan recognition. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 32–38, 1986.

[9] N. Lesh, C. Rich, and C. Sidner. Collaborating with focused and unfocused users. In *Proceedings of the 8th International Conference on User Modeling*, 2001.