Design and Development of
an NIDS Evasion Test Tool with GUI

Chung-Huang Yang[1], Chih-Hung Lin[2], Li-Chou Juan[2], Jain-Shing Wu[2], and Tai-Yun Huang[1]

[1]Graduate Institute of Information and Computer Education
National Kaohsiung Normal University
Kaohsiung 802, TAIWAN

[2]Networks & Multimedia Institute, Institute for Information Industry
7F, No. 218, Sec. 2, Dunhua S. Rd., Taipei 106, TAIWAN

Abstract

Evasion techniques are modifications made to prevent detection by a network intrusion detection system (NIDS). Evasion attacks are one of the most fundamental challenges in NIDS, as they undermine the NIDS in its basic perception of the network stream. If an NIDS interprets traffic in a different fashion than do the involved endpoints, the NIDS cannot reliably detect attacks.

This work describes a developed tools for assessing the degree to which NIDSs are vulnerable to (or are robust against) different forms of evasions. The automated tool is built with a graphic user-interface (GUI) on Linux platform so that it could be easily used. We conducted an evaluation of our tool against the popular open-source Snort NIDS.

Keywords: Intrusion detection, evasion, stealthy, security testing, protocol scrubber

1 Introduction

An intrusion detection system (IDS) [2] analyzes one or more streams of events and looks for the manifestations of attacks. Network-based IDSs (NIDSs) analyze network packets and are mostly based on misuse detection approach. There are two primary approaches for NIDSs to analyzing events to detect attacks: misuse detection and anomaly detection [2]. NIDSs with misuse detection rely on models, so-called “signatures”, of attacks to identify the manifestation of intrusive behavior. Each pattern of event corresponding to known attack is called a signature and such misuse detectors are very effective to detect attacks without generating an overwhelming number of false alarms. For example, Snort [1, 8] is the leading open-source NIDS in the world, which can perform a variety of traffic logging and analysis functions on networks.

Evasion attacks [3, 6, 7, 10] are one of the most fundamental challenges in network intrusion detection, as they undermine the NIDS in its basic perception of the network stream. If a NIDS interprets traffic in a different fashion than do the involved hosts, the NIDS cannot reliably detect attacks. While these attacks do not necessarily increase the vulnerability of the involved hosts, they do significantly deteriorate the NIDS's ability to detect attacks [4, 5]. Currently, no comprehensive test-suite is available to the public to systematically probe a NIDS to assess its resilience against such attacks. Most current NIDS address evasion attacks in at least a limited fashion. The sophistication of these implementations varies widely, however, and due to the black-box nature of many of these systems, it is difficult to get a good picture of what a NIDS is actually doing to resist evasion attacks.

In this paper we present our effort in the design and implementation of an automated tools on Linux platform for assessing the degree to which NIDSs are vulnerable to (or are robust against) different forms of evasions. Our tool is built with a graphic user-interface (GUI) on Linux platform so that it could be easily used. We conducted an evaluation of our tool against the popular Snort NIDS.

2 Evasion Techniques

Evasion or exploit mutation [6, 7, 11] is a general term comprising a broad range of techniques to modify and obfuscate an attack again a vulnerable service. For example, the victim host could accept a packet that is ignored by the NIDS, such as the attacker sends extra packets with the same sequence number as a previous packet but with different data. The NIDS might drop the packet because the sequence number was already used while the victim host (depending on the OS) would accept and process the packet, replacing a previous substring with one that turns the entire message into a hack. Many other evasion attacks are possible.

As shown in Fig. 1, evasion techniques could proceed as follows:

(1) The signatures are put into the mutation engine.

(2) The mutation engine will mutate the signatures according to the mutation mechanisms.

(3) The mutation engine sends the mutant to the network.

(4) The mutant evades the NIDS's detection and reaches victim host.

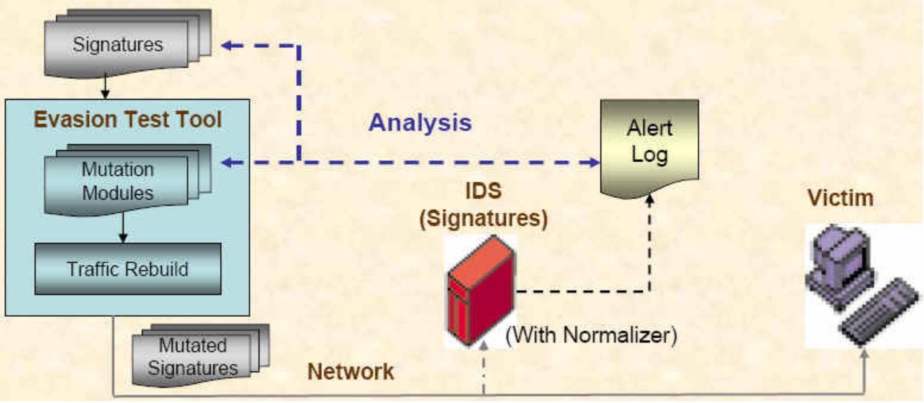
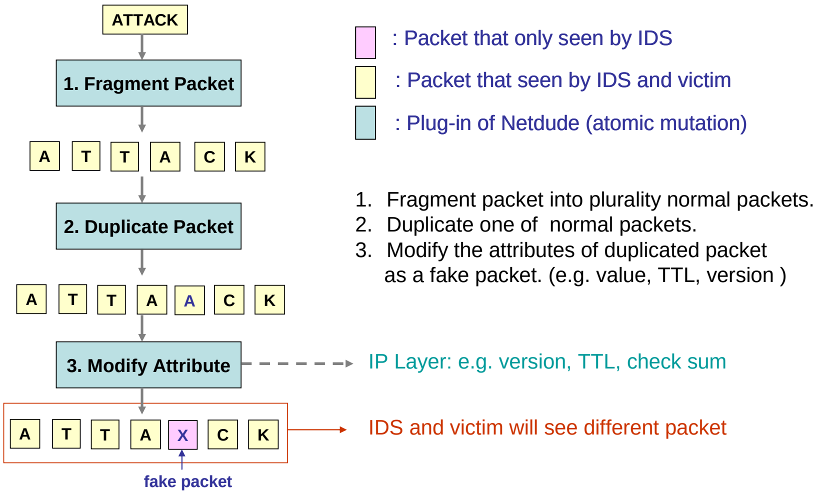


Fig. 1. IDS Circumvention Framework

Many evasion techniques are known [6, 9] and can operate at several different layers of the classic OSI networking model. A victim host can accept a packet that an IDS rejects. An IDS that mistakenly rejects such a packet misses its contents entirely. This condition can also be exploited, this time by slipping crucial information past the IDS in packets that the IDS is too strict about processing. These packets are "evading" the scrutiny of the IDS.

An example of IP evasion [6] is illustrated at Fig. 2. An attacker confronts the NIDS with a stream of one character packets in which one of the normal packets will be duplicated and accepted only by the NIDS. As a result, the NIDS and the victim host reconstruct two different strings.


ATTACK
1. Fragment Packet
A T T A C K
2. Duplicate Packet
A T T A A C K
3. Modify Attribute
A T T A X C K
fake packet
: Packet that only seen by IDS
: Packet that seen by IDS and victim
: Plug-in of Netdude (atomic mutation)
IP Layer: e.g. version, TTL, check sum
IDS and victim will see different packet

1. Fragment packet into plurality normal packets.
2. Duplicate one of normal packets.
3. Modify the attributes of duplicated packet as a fake packet. (e.g. value, TTL, version)

Fig. 2. An Example of IP Evasion

An example of TCP evasion [3] is illustrated at Fig. 3. With TCP, the SYN control signal for connection establishment is delivered reliably, but the RST (abrupt termination) signal is not. The rules applied by receiving hosts to determine whether a particular RST signal is valid or not vary across different OSs, which the NIDS likely cannot track. This allows attackers to deliberately violate the TCP protocol in a way that victim host will handle differently than the NIDS.

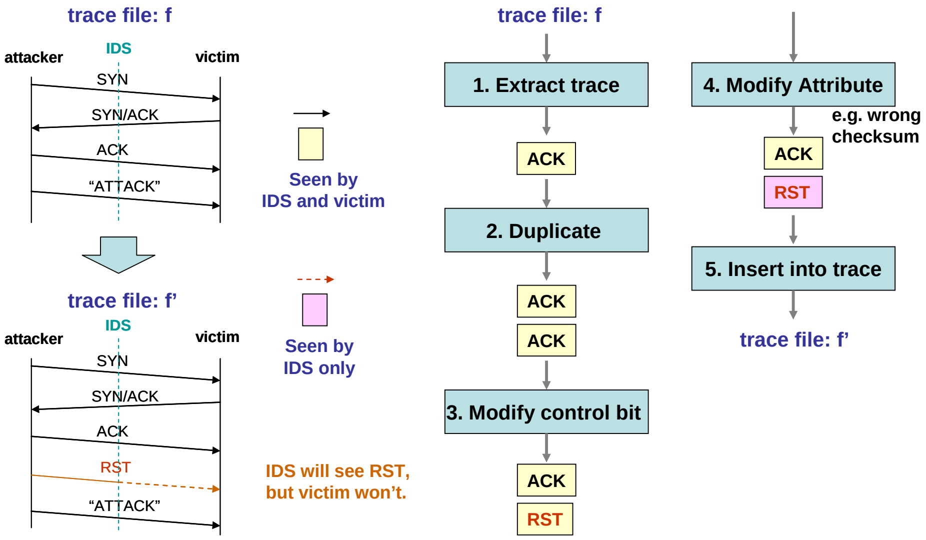
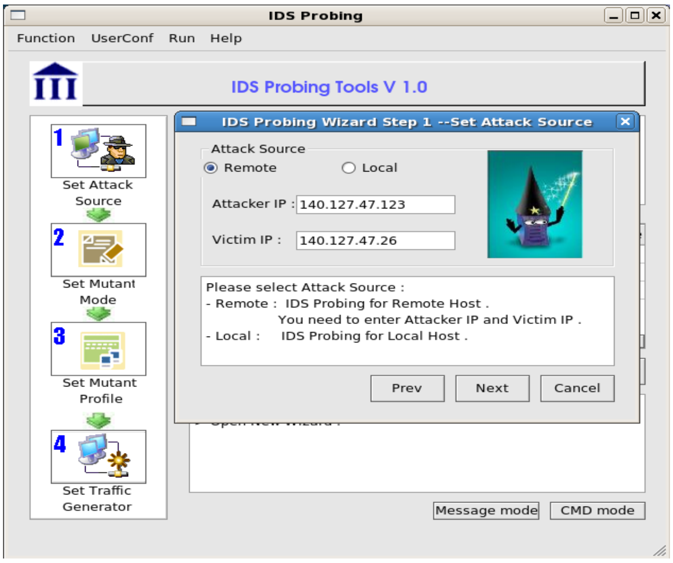


Fig. 3. An Example of TCP Evasion

3 Implementation of an Evasion Tool

We developed an evasion tool on FedoaCore Linux platform. Both IP layer mutation (such as bad header fields, bad IP options, overlapping fragments) and TCP layer mutation (such as malformed header fields, bad TCP options, TCP teardown, TCP stream reassembly) are supported.

Fig. 4 shows the user-friendly wizard of our tool as the first step to set attack source.



IDS Probing Wizard Step 1 --Set Attack Source

Attack Source

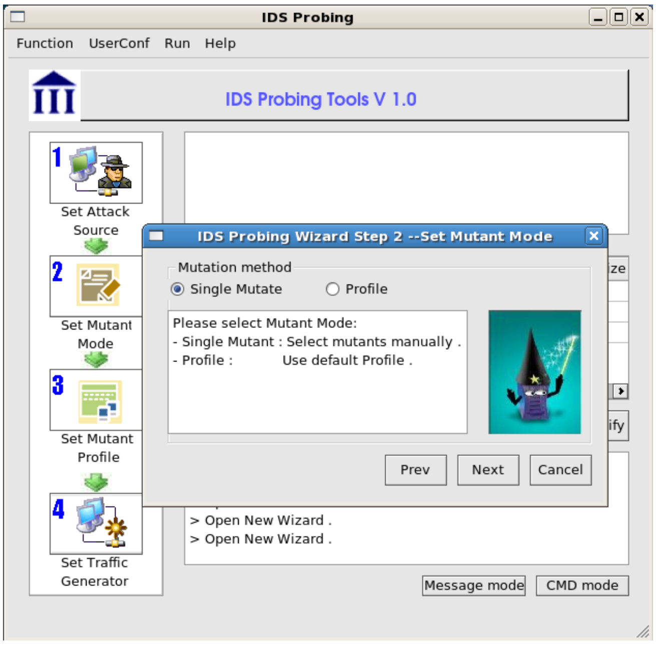
● Remote ○ Local

Attacker IP : 140.127.47.123

Victim IP : 140.127.47.26

Please select Attack Source :
- Remote : IDS Probing for Remote Host .
You need to enter Attacker IP and Victim IP .
- Local : IDS Probing for Local Host .

Fig. 4. The Developed IDS Evasion Tool



IDS Probing Wizard Step 2 --Set Mutant Mode

Mutation method

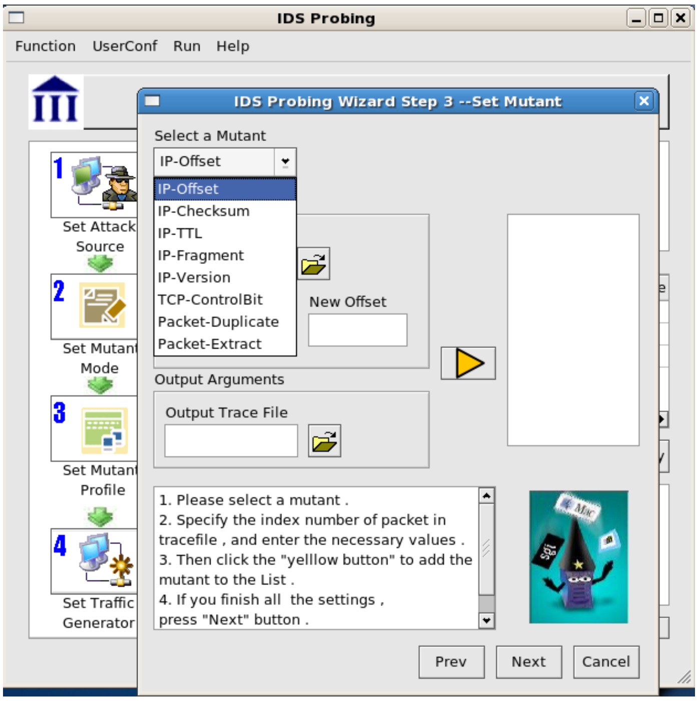
● Single Mutate ○ Profile

Please select Mutant Mode:
- Single Mutant : Select mutants manually .
- Profile : Use default Profile .

> Open New Wizard .
> Open New Wizard .

Fig. 5. Selection of Single Mutant or Profile

Mutant mode can be selected manually or by a profile-based automation as the second step, shown in Fig. 5. Each installed mutant is selected with on-line help hint to indicate what additional parameters are needed, this is shown in Fig. 6.



1. Please select a mutant .
2. Specify the index number of packet in tracefile , and enter the necessary values .
3. Then click the "yelllow button" to add the mutant to the List .
4. If you finish all the settings , press "Next" button .

Fig. 6. Pre-Installed Techniques for Mutant Generation

4 Experiment

In order to test the effectiveness of our tool, an experiment of the tool again Snort was conducted. The objective of the experiment was to determine whether the developed tool is capable of automatically generating mutant that can evade Snort. Table 1 presents the evaluation results.

Evasion attack	Original packet	Modified packet
1. Duplicate TCP-payload-carrying packet in a TCP flow	Detected	Detected
2. Duplicate TCP-payload-carrying packet (Modify first or second)	Detected	Detected
3. Duplicate TCP-payload-carrying packet and modify (IP-TTL, TCP-Checksum, TCP-Offset, IP-Version) value	Detected	Detected
4. Fragment IP packets in different offset	Detected	Detected
5. Fragment an IP packet. Duplicate a fragment and change its contents (TTL)	Detected	Undetected
6. Fragment an IP packet. Duplicate a fragment and change its payload	Detected	Detected

Table 1. The Developed Tool Probing for Snort

5 Conclusions

NIDSs have been widely deployed and the evaluation of effectiveness of them is very important. NIDSs rely on signatures to recognize malicious behavior. This paper presented a developed GUI tool for the black-box evaluation of NIDS. Our tool generates, manually or automatically, mutant for testing the quality of preprocessor used by NIDS.

Acknowledgement

This work was supported by a research grant NSC 94-3114-P-006-001-Y from the National Science Council of Taiwan.

References

[1] R. Alder, J. Babbin, A.R. Baker, A. Doxtater, J. C. Foster, and M. Rash, *Snort 2.1 Intrusion Detection*, 2nd Edition, Syngress Publishing, Inc., 2004.

[2] R. Bace and P. Mell, “Intrusion Detection Systems,” *NIST Special Publication 800-31*, Nov. 2001. http://csrc.nist.gov/publications/nistpubs/800-31/sp800-31.pdf

[3] M. Handley, V. Paxson, and C. Kreibich, “Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics,” *10th USENIX Security Symposium*, 2001.

[4] P. Mell et al., "An Overview of Issues in Testing Intrusion Detection Systems," *NIST Interagency Report 7007*, June 2003.

[5] NSS Group, *Network IPS Testing Procedure (V4.0)*, 2006. http://www.nss.co.uk/certification/ips/nss-nips-v40-testproc.pdf

[6] T. Ptacek and T. Newsham, “Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection,” Technical Report, Secure Networks, Inc., January 1998.

[7] U. Shankar and V. Paxson, “Active Mapping: Resisting NIDS Evasion without Altering Traffic," *Proc. 2003 IEEE Symposium on Security and Privacy*, 2003, pp. 44-61. http://www.icir.org/vern/papers/activemap-oak03.pdf

[8] Snort, http://www.snort.org/

[9] G. Vigna, W. Robertson, and D. Balzarotti, "Testing Network-based Intrusion Detection Signatures Using Mutant Exploits," *Proceedings of the ACM Conference on Computer and Communication Security (ACM CCS)*, pp. 21-30, 2004. http://cs.ucsb.edu/~seclab/projects/sploit/2004_vigna_robertson_balzarotti_CCS

04.pdf

[10] D. Watson, M. Smart, and G. R. Malan, “Protocol Scrubbing: Network Security Through Transparent Flow Modification,” *IEEE/ACM Trans. Networking*, Vol. 12, No. 2, April 2004, pp. 261-273.

[11] Wikipedia, Intrusion Detection System Evasion Techniques, http://en.wikipedia.org/wiki/Intrusion_detection_system_evasion_techniques