Intrusion Detection System

Objectives
• To know what is Intrusion Detection system and why it is needed.
• To be familiar with Snort IDS/IPS.

What Is Intrusion Detection?
Intrusion is defined as “the act of thrusting in, or of entering into a place or state without invitation, right, or welcome.” When we speak of intrusion detection, we are referring to the act of detecting an unauthorized intrusion by a computer on a network. This unauthorized access, or intrusion, is an attempt to compromise, or otherwise do harm, to other network devices.

• An IDS is designed and used to detect attacks or unauthorized use of systems, networks, and related resources, and then in many cases to deflect or deter them if possible. Like firewalls, IDSes can be software-based or can combine hardware and software in the form of preinstalled and preconfigured stand-alone IDS devices.

• IDSes can detect and deal with insider attacks as well as external attacks, and are often very useful in detecting violations of corporate security policy and other internal threats.

Classification of IDSs:
IDSes are classified by their functionality and are loosely grouped into the following three main categories:

Network-based intrusion detection system (NIDS)
- Operation of the NIDS’s NIC in promiscuous mode is necessary to protect your network.
- The use of multiple NIDS within a network is an example of a defense-in-depth security architecture.
Host-based intrusion detection system (HIDS)
- HIDS protects only the host system on which it resides, and its network card operates by default in nonpromiscuous mode.
- Another advantage of HIDS is the capability to tailor the ruleset very finely for each individual host (customized rules).

Distributed intrusion detection system (DIDS)
- The standard DIDS functions in a Manager/Probe architecture. NIDS detection sensors are remotely located and report to a centralized management station.
- The network transactions between sensor and manager can be on a private network, as depicted, or the network traffic can use the existing infrastructure. When using the existing network for management data, the additional security afforded by encryption, or virtual private network (VPN) technology, is highly recommended.
Why Are Intrusion Detection Systems Important?

- No network is too small to be left unprotected. If a hacker can use your computer, he will. Multiple computers operating in concert perform DDoS attacks. Hacker masters need zombies.
- Logs from ID Ses are an important part of computer forensics and incident handling efforts.
- ID Ses keep you informed of your network’s health and security.
- ID Ses can detect failed administrator login attempts and recognize password-guessing programs.
- Inline ID Ses can halt active attacks on your network while alerting administrators to their presence.
- You can use ID Ses to identify vulnerabilities and weaknesses in your perimeter protection devices; in other words, firewalls and routers.
- You can use ID logs to enforce company policy.
- You can verify firewall rules and router access lists regularly for functionality.
- Buffer overflow attacks represent a large percentage of today’s computer exploits.
- Backdoors and Trojans are remote control programs that are malicious code designed to take control of your computer. Snort can detect the communications of these Trojans and alert you to their presence.
- E-mail servers are prime targets for intrusions. They must be accessible from the Internet, and thus are vulnerable to attack.
- You can use ID Ses for a variety of functions in addition to detection of intrusions, including monitoring database access, monitoring DNS services, protecting your e-mail server, and monitoring corporate policies.

What About Intrusion Prevention?

It may be useful to clarify the difference between inline-IDS and IPSes. An inline IDS is deployed at a choke point in one’s network topology, forcing all traffic to flow through the inline IDS device. This allows the IDS to selectively drop traffic that matches its signature base of malicious attack traffic. An IPS, on the other hand, generally takes an even more active stance than an inline IDS. Most IPSes are deployed in an inline configuration, but not all are. IPSes deployed in the less-common one-armed configuration generally attempt to prevent malicious traffic from continuing by issuing TCP resets to one or both participants in the conversation. However, this is less effective than being inline and simply dropping, disrupting, or otherwise controlling the traffic. IPSes may optionally take additional action such as dynamically adding the attacking machine to block lists, performing network block ownership lookup, and in some cases scanning the attacking system back. Active response that includes blocking or session reset is generally accepted, though false positives in this have a greater network impact than IDS alerts. However, strikeback is still greatly controversial, not to mention legally ambiguous, and so not generally implemented.
Lab Experiment

Requirements:
1. We use in this experiment Snort as IDS you can get the latest version from www.snort.org.
   Snort is an open source network intrusion prevention and detection system (IDS/IPS) developed by Sourcefire. Combining the benefits of signature, protocol and anomaly-based inspection, Snort is the most widely deployed IDS/IPS technology worldwide. With millions of downloads and over 250,000 registered users, Snort has become the de facto standard for IPS.
2. Setup snort IDS on one machine you can use Linux systems, Solaris, windows or Mac OS. (suppose we want to run it over windows)
3. Other machines may be act as attackers.

Procedures:

Snort as sniffer:
1. Setup WinPcap on your system then install the latest version from snort that you get from www.snort.org.
2. Simply open cmd and go to snort path (e.g.: c:\snort\bin if you install it on driver c).
3. From c:\snort\bin prompt you can use snort; to run it as sniffer type snort –dev -i 2; sample output shown in the figure below
4. to exit from snort type ctrl+c; then the summary of the sniffed packets will appear as shown in the figure.
   (you can list switches of snort by type snort.exe)
Snort as IDS:

Open snort.conf file (you will find it in c:\snort\etc) and do the following changes to adopt your conf file to windows:

1. Set the variables for your network
   - You must change the following variables to reflect your local network; define your network in any of the following forms, A good start may be "any"

```
var HOME_NET $eth0_ADDRESS
var HOME_NET [10.1.1.0/24,192.168.1.0/24]
var HOME_NET any
```
   - Set up the external network addresses as well. A good start may be "any"

```
var EXTERNAL_NET any
```
   - Configure your server lists. This allows snort to only look for attacks to systems that have a service up. Why look for HTTP attacks if you are not running a web server? This allows quick filtering based on IP addresses. Take the following examples that define DNS server and SMTP server for the same your home network address; (change it as your server ip address):

```
var DNS_SERVERS $HOME_NET
var SMTP_SERVERS $HOME_NET
```

2. Configure dynamic loaded libraries
   - Change the following line

```
dynamicpreprocessor directory /usr/local/lib/snort_dynamicpreprocessor/
```
   - with

```
dynamicpreprocessor directory C:\Snort\lib\snort_dynamicpreprocessor
```
   - and this line

```
dynamicengine /usr/local/lib/snort_dynamicengine/libsf_engine.so
```
with
dynamicengine C:\Snort\lib\snort_dynamicengine/sf_engine.dll

Why do you think this changes must be done?
because this paths for Linux systems.

3. Configure preprocessors

4. Configure output plugins
   - Change the following two lines
     include classification.config
     include reference.config
   with
     include c:\snort\etc\classification.config
     include c:\snort\etc\reference.config
   - add this line to write the log details in
     output alert_fast : xx.ids
   - **Note**: you must create file xx.ids in log folder in c:\snort

5. Add any runtime config directives

6. Customize your rule set
   - Modify the path of rules folder as follows
     var RULE_PATH ..\rules
     var RULE_PATH c:\snort\rules
   - download rules from snort.org and add it in rules folder
   - each rule you want to activate you must add this line for it; as an example if we want to activate icmp-info rule we add:
     include $RULE_PATH/icmp-info.rules
   - **Note**: rules form snort.org contains huge rules that you can add to detect intrusion on your system:

**Note**: snort.conf file contains a sample snort configuration, you can built your own in the same structure as your network required as mentioned in the above steps with different name.

**Example One:**
1- Do the previous steps from 1 to 5
2- Add the following lines in snort.conf file (remove comment if this line found and commented)
   include $RULE_PATH/icmp.rules
   include $RULE_PATH/icmp-info.rules
3- form console; run the following command
   snort -c c:\snort\etc\snort.conf -l c:\snort\log -i 2
4- try to ping your machine where snort installed.
5- [Get result] go to log folder and open xx.ids file; you will find who are pinging your machine and how many times he do this.

**Example Two: Built your your custom rules**

1- Create new file with iug.rules name in rules folder where snort installed
2- Open the file and add the following line:
   `alert tcp any any -> any any (content:"www.iugaza.edu.ps" ; msg:"someone browsing IUG site now ";sid:10000020;rev:1;)
   
3- add the following line to snort.conf file to activate your rule:
   `include $RULE_PATH\iug.rules
   
4- form console; run the following command
   `snort -c c:\snort\etc\snort.conf -A console -i 2
   
5- open www.iugaza.edu.ps
6- [Get result] the logs now will shows in the console as the command says.

For more information about snort it is recommended to read from

**O'Reilly Snort Cookbook**, By Jacob Babbin, Simon Biles, Angela D. Orebaugh

**References:**

1- www.snort.org
2- **O'Reilly Snort Cookbook**, By Jacob Babbin, Simon Biles, Angela D. Orebaugh
3- **Snort IDS and IPS Toolkit**, Jay Beale, Andrew R. Baker and Joel Esler.