Intrusion Detection

- Intrusion Detection (ID) is a set of techniques and methods that are used to detect suspicious activity both at the network and host level.
- Intruders have signatures, like computer viruses, that can be detected using specialized software and hardware called an Intrusion Detection system (IDS).
- An IDS will try to find packets that contain any known intrusion-related signatures or anomalies related to Internet protocols.
- Based upon a set of signatures and rules, an IDS is able to **capture** and **log** suspicious activity and generate **alerts**.
- Anomaly-based intrusion detection is based on packet anomalies present in the protocol header fields.
- In some certain cases these methods produce better results compared to signature-based IDS.
- Usually an intrusion detection system captures data from the network and applies its rules to that data or detects anomalies in it.
- Snort is primarily a rule-based IDS, however input plug-ins are available to detect anomalies in protocol headers.
- There are two approaches to intrusion detection: **host-based** and **network-based** ID systems.
- Current strategy is to move towards a unified approach combining the two approaches.
- In either case, these products look for **attack signature**s, which are specific patterns indicating malicious or suspicious intent.
- A network-based IDS looks for these patterns in network traffic while a host-based IDS looks for attack signatures in log files.
- Each approach has its strengths and weaknesses, and each is complements the other. An effective intrusion detection system is one that employs both technologies.

Network-Based Intrusion Detection

- Network-based intrusion detection systems use raw network packets as the data source. This process involves looking at the packets on the network as they pass by some sensor.

- A NIDS typically utilizes a sensor that is a network adapter running in promiscuous mode to monitor and analyze all traffic in real-time as it travels across the network.

- Packets are considered to be of interest if they match a specified signature. The signature recognition module uses four primary types of signatures:

    - Pattern, expression or bytecode matching (string signatures)
    - Frequency or threshold crossing (statistical detection)
    - Port signatures
    - Header condition matching

- String signatures look for a text string that indicates a possible attack. An example string signature for UNIX might be **"cat "+ +" > /.rhosts",** which if successful, might cause a UNIX system to become extremely vulnerable to network attack.

- To refine the string signature to reduce the number of false positives, it may be necessary to use a compound string signature. A compound string signature for a common Web server attack might be "**cgi-bin**" AND "**aglimpse**" AND "**IFS**".

- Port signatures monitor connection attempts to well-known, frequently attacked ports. Examples of these ports include telnet (TCP port 23), FTP (TCP port 21/20), SUNRPC (TCP/UDP port 111), and IMAP (TCP port 143).

- Header signatures watch for header field combinations in packet headers that are not normal, that is so say not used for typical network traffic.

- A good example is Winnuke, where a packet is destined for a NetBIOS port and the Urgent pointer, or Out Of Band pointer is set. This resulted in the "blue screen of death" for older Windows systems.

- Another well-known header signature is a TCP packet with both the SYN and FIN flags set, signifying that the requestor wishes to start and stop a connection at the same time.

Advantages of Network-based IDS

- Many network administrators deploy network-based intrusion detection when using an IDS for the first time due to its low cost of ownership and rapid response times.

- Some of the main reasons that make network-based intrusion detection a critical component of overall network security design are:

1. **Lowers cost of ownership**

    - Network-based IDS allow strategic deployment at critical access points for viewing network traffic destined to multiple systems.

2. **Detects attacks that host-based systems miss**

    - Network-based IDS examine all packet headers for signs of malicious and suspicious activity.

    - Host-based IDS do not see packet headers, so they cannot detect these types of attacks.

3. **More difficult for an attacker to remove evidence**

    - Network-based IDS use live network traffic for real-time attack detection. Therefore, an attacker cannot remove the evidence.

    - Captured data includes not only the method of attack, but information that may help lead to identification and prosecution.

    - Since many hackers understand audit logs, they know how to manipulate these files to cover their tracks, frustrating host-based systems that need this information to detect an intrusion.

4. **Real-time detection and response**

    - Network-based IDS detect malicious and suspicious attacks **as they occur**, thus providing immediate notification and response.

5. **Detects unsuccessful attacks and malicious intent**

    - Network-based IDS add valuable data for determining malicious intent.

    - A network-based IDS placed outside of a firewall can detect attacks intended for resources behind the firewall, even though the firewall may be rejecting these attempts.

6. **Operating system independence**

    - Network-based IDS are not dependent on host operating systems as detection sources.

Host-Based Intrusion Detection

- Host-based intrusion detection started in the early 1980s before networks were as prevalent, complex and interconnected as they are today.
- In that simpler environment, it was common practice to review audit logs for suspicious activity. Intrusions were sufficiently rare that after-the-fact analysis proved adequate to prevent future attacks.
- While host-based intrusion detection systems are not as fast as their network counterparts, they do offer advantages that the network-based systems cannot match.
- These include stronger forensic analysis, a close focus on host-specific event data and lower entry-level costs.
- The following are some strengths of host-based intrusion detection:

1. **Verifies success or failure of an attack**
    - Since host-based IDS use logs containing events that have actually occurred, they can measure whether an attack was successful or not with greater accuracy and fewer false positives can network-based systems.
    - In this regard, host-based IDS make an excellent complement to network-based intrusion detection, with the network component providing early warning and the host component providing verification of whether an attack was successful or not.

2. **Monitors specific system activities**
    - Host-based IDS monitor user and file access activity, including file accesses, changes to file permissions, attempts to install new executables and/or attempts to access privileged services.
    - Host-based systems can monitor changes to key system files and executables.
    - Attempts to overwrite vital system files, or to install Trojan horses or backdoors, can be detected and stopped. Network-based systems can miss this kind of activity.

3. **Detects attacks that network-based sensor cannot see**

    - Host-based systems can detect attacks that cannot be seen by network-based products.

    - For example, attacks from the keyboard of a critical server do not cross the network, and so cannot be seen by a network-based intrusion detection system.

4. **Well-suited for encrypted and switched environments**

    - Since host-based systems reside on various hosts throughout an enterprise, they can overcome some of the deployment challenges faced by network-based intrusion detection in switched and encrypted environments.

5. **Requires no additional hardware**

    - Host-based intrusion detection resides on existing network infrastructure, including file servers, Web servers, and other shared resources.

    - This efficiency can make host-based systems very cost effective because they do not require another box on the network that requires addressing, maintenance, and management.

6. **Lower cost of entry**

    - While network-based intrusion detection systems can offer wide coverage for little effort, they are often expensive.

    - Deploying a single intrusion detection system can cost in the range of tens of thousands of dollars.

    - Host-based intrusion detection systems, on the other hand, are often priced in the hundreds of dollars for a single agent and can be deployed by a customer with limited initial investment.

Network and Host-Based IDS Response Options

- Once an attack has been detected, the IDS' response module provides a variety of options to notify, alert and take action in response to the attack.
- Response capabilities for threats and attacks are crucial for any intrusion detection system.
- These responses vary by product, but typically involve administrator notification, connection termination and/or session recording for forensic analysis and evidence collection.
- Most network- and host-based IDS share common threat and attack response options.
- The responses fall into three general categories: **Alert**, **Log**, and **Active** responses.

Alerts

- Alerts are any sort of user notification of an intruder activity. When an IDS detects an intrusion, it has to inform security administrator about this using alerts.
- Alerts may be in the form of pop-up windows, logging to a console, sending e-mail and so on.
- Alerts are also stored in log files or databases where they can be viewed later on by security analysts.
- And IDS such as Snort can generate alerts in many forms and are controlled by output plug-ins.
- Snort can also send the same alert to multiple destinations. For example, it is possible to log alerts into a database and generate SNMP traps simultaneously.
- Some plug-ins can also modify firewall configuration so that offending hosts are blocked at the firewall or router level.

Logs

- The log messages are usually saved in file. For example, by default Snort saves these messages under the **/var/log/snort** directory.
- Log messages can be saved either in text or binary format. The binary files can be viewed later on using various tools including tcpdump.
- A tool called Barnyard is also available now to analyze binary log files generated by Snort. Logging in binary format is faster because it saves some formatting overhead. In high-speed IDS implementations, logging in binary mode is necessary.

Active Response

- Active Response is a mechanism in an IDS that provides the IDS with capability to respond to an attack when it has been detected.
- There are two methods that the IDS can take to respond to an attack. The first method is **Session disruption**, and the second is **Filter rule manipulation**.
- Session disruption is a more frequently used active response method because of the ease of its implementation.
- Depending on the type of session established, UDP or TCP, an IDS that is configured for session disruption can reset or tear down the established connection.
- This does not prevent the attacker from launching additional attacks, but it does prevent the attacker from causing any further damage in conjunction with the "broken" session.
- With session disruption the IDS uses different methods for breaking the connection depending on the type of traffic it detects.
- If an attacker uses TCP sessions, they are reset by RST packet that is sent to reset one or both hosts in a session from the IDS.
- In the case of UDP, a session can be terminated by sending various ICMP packets to the host from the IDS.
- ICMP packets are sent to a host initiating a UDP connection to inform the sender that a requested port/host is unavailable.
- The reason ICMP packets are sent to a UDP stimulus is UDP does not have the capability to report errors, so ICMP is used to assist.
- Snort use this normal process to send a spoofed ICMP packet to the host initiating the connection in an attempt to fool the host in to thinking that the host is unavailable.
- The Filter rule manipulation mechanism works by modifying the access control list (ACL) on a firewall or router.
- Filter rule manipulation block the IP of the attacker preventing any further attacks. This option should be used with extreme care, because an attacker can use it to launch a Denial of Service the network.
- If an attacker used the IP address of a partner or a trusted network they could spoof the address. When the IDS detects the attack and responds by blocking the network, it would in effect be a DoS.

- The following table summarizes these.

Categories	Network-Based IDS	Host-Based IDS
Alert	Alarm to Console E-Mail Notification SNMP Trap View Active Session	Alarm to Console E-Mail Notification SNMP Trap
Log	Log Summary (Reporting) Log Raw Network Data	Log Summary (Reporting)
Active	Kill Connection (TCP Reset) Re-Configure Firewall User Defined Action	Terminate User Login Disable User Account User Defined Action

Analysis of Network Data and Events of Interest

- There is a vast amount of data (observable or otherwise) available on a network but a significant amount of that data is irrelevant as far as intrusion detection is concerned.
- An intrusion analyst is only interested in that data which is useful, that is, Events of Interest (EOI).
- EOIs allow the analysts to focus their attention on those detects that pose a potential threat to the network and determine the **severity** of a particular event.
- It is important to identify false positives and false negatives in the overall data set.
- False positives are "false alarms." These detects match only some of the criteria for indicators of possible intrusion.
- False positives tend to use up incident handling resources and in the long run tend to desensitize the analyst to more serious intrusion attempts.
- False negatives are the actual intrusions and intrusion attempts that went undetected.
- These can allow an attacker to establish a beachhead on the network from which to launch further attacks before we start to react and deploy countermeasures.
- Given the limited computing and personnel resources in most organizations, analysts cannot collect, store, and analyze all possible events.
- Therefore, analysts tend to focus their collection efforts on events that they might pose a threat to the network. These are the Events of Interest (EOI).
- Narrowing the data-gathering window helps reduce the false alarms or false positives, but increases the chance of missing an EOI.
- One way to help ensure that an EOI is not missed is to compare suspicious events against a dictionary of known attacks or attackers.
- Another way to widen our field of vision is to monitor for short-term variations in system or user traffic profiles.

Severity

- Severity is defined by the criticality of the target, the lethality of the attack, and the effectiveness of system and network countermeasures.
- The impact of the event is calculated by the analyst; delays in detection and reaction can increase severity and impact.
- The severity of the attack is determined by evaluating a set of four variables:
    - Criticality of the victim host (critical network services)
    - Lethality of the attack (achieve root access?)
    - System countermeasures (all patches applied, etc)
    - Network countermeasures (firewalls, perimeter, etc)
- Each of the variables above is assigned a numerical value based on a scale of 1 (low), to 5 (high).
- The overall severity of the attack is then calculated as follows:

Severity = (criticality + lethality) – (System + Network countermeasures)

Criticality

- The main issue to consider for this parameter is how critical the target machine is to the rest of the network.
- We can assign points on a five point scale as follows:
    - 5 points: firewall, DNS server, core router
    - 4 points: e-mail relay/exchanger
    - 2 points: user Unix/Win32 desktop system
    - 1 point: Other older OS's.
- A compromised desktop system will result in time and work being lost.
- In addition, that machine can now be used to launch attacks other systems.
- An organization's Domain Name System (DNS) server or electronic mail relay being compromised is a much more serious problem.
- An attacker can take over a site's DNS server and manipulate trust relationships and thereby compromise most or all of a site's systems.

Lethality

- The main issue to consider for this parameter is how likely is the attack to succeed and do damage.
- We can assign points on a five point scale as follows:
    - 5 points: attacker can gain root access the network
    - 4 points: denial of service
    - 3 points: user access (sniffed password)
    - 2 points: Unauthenticated Connection (Null session)
    - 1 point: Very unlikely to succeed (Code Red on Apache)
- The attack software is either application or operating system specific.
- A Macintosh desktop system is not vulnerable to NULL session attacks but a Windows machine is quite vulnerable to this.
- Gaining NULL session access to a Windows system is the number one method for hackers to enumerating information about that machine.
- From a NULL session hackers can call APIs and use Remote Procedure calls to enumerate information.
- These techniques can, and will provide information on passwords, groups, services, users and even active processors.
- A Linux box running an unpatched printer daemon might quickly become compromised with the attacker acquiring full root access.

System Countermeasures

- The main issue to consider for this parameter is how up to date the OS patches from the vendor have been applied and how well the services on that system are protected.
- Higher scores are awarded for systems that are running a modem operating system, that are current with all patches, and are using additional security measures such as TCP wrappers, secure shell, and personal firewalls.
- We can assign points on a five point scale as follows:
    - 5 points: modern operating system, all patches, and added security (see above)
    - 3 points: older operating system, some patches missing
    - 1 points: No wrappers/allows fixed passwords

Network Countermeasures

- The main issue to consider for this parameter is how well protected the network is using firewalls, perimeters, etc.
- Network countermeasures are the first line of defense whereas system countermeasures are the last line of defense.
- Higher scores are awarded for networks that deploy very restrictive firewalls with only one way in or out.
- We can assign points on a five point scale as follows:
    - 5 points: validated restrictive firewall (one way in or out)
    - 4 points: restrictive firewall (some external connections allowed)
    - 2 points: permissive firewall (allows the attack to go through)
- In any network the network countermeasures are the most probed and tested defenses.

Example

Detect : Redhat 7.0 lprd Overflow

- The following trace was captured on a sensor running on a subnet assigned to a school in an educational institute.

- The following is the entry from the Snort alert file:

  [**] [1:302:1] EXPLOIT redhat 7.0 lprd overflow [**]
  [Classification: Attempted Administrator Privilege Gain] [Priority: 10]
  10/12-02:56:15.696382 195.61.80.253:3069 -> x.x.x.x:515
  TCP TTL:42 TOS:0x0 ID:45360 IpLen:20 DgmLen:475 DF
  AP Seq: 0x8A934D1D Ack: 0xAAE94C7A Win: 0x7D78 TcpLen: 32**
  TCP Options (3) => NOP NOP TS: 22117945 124961620

Source of Trace

- This trace was captured on a sensor, running on a publicly accessible subnet. The subnet is connected to the backbone via a router. The IDS was running on the server that was targeted for this exploit.

- The server also provides secondary web services, print services, Samba, and NFS services.

Detect was generated by:

- Snort using the default rule sets provided with the package. The rule activating this alert is found in the "exploit.rules" file.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 515 (msg:"EXPLOIT LPRng
overflow"; flags: A+; content: "|43 07 89 5B 08 8D 4B 08 89 43 0C
B0 0B CD 80 31 C0 FE C0 CD 80 E8 94 FF FF FF 2F 62 69 6E 2F 73 68
0A|"; reference:bugtraq,1712; classtype:attempted-admin; sid:301;
rev:1;)
```

Probability that the source address was spoofed

- The exploit involves causing a buffer overflow on the target system, establishing a TCP connection to the victim host and executing programs so the IP address cannot be spoofed.

Description of attack

- This is an exploit that specifically targets RedHat 7.0 systems.
- A popular replacement software package to the BSD ***lpd*** printing service is called **LPRng** contains at least one software defect known as a "format string vulnerability" which may allow remote users to execute arbitrary code on vulnerable systems.

Attack mechanism

- The source code for this exploit is available as: "**rdC-LPRng.c**" from any number of sites that provide exploits.
- This particular program will allow a user to construct a buffer that will insert a string in the missing format and overwrite addresses in the printer daemon and cause it to crash.
- The exploit code then inserts its own shell code in the code space of the daemon and thus execute can any program.
- In the trace above we will observe this code at the very end of the payload. The string "**/bin/sh**" will give the attacker a root console and thus the means to execute any program with full root privileges. The destination port is TCP port 515 the print spooler.

Correlations

- This vulnerability has been widely reported in many references including:

http://lists.suse.com/archives/suse-security/2000-Sep/0259.html

http://www.redhat.com/support/errata/RHSA-2000-065-06.html

Evidence of active targeting

- This is most certainly active targeting. The server was targeting with the express purpose of compromising it and acquiring root access.

Severity

- The overall severity of the attack is then calculated as follows:

  Severity = (criticality + lethality) – (System + Network countermeasures)

- Criticality = 3. The target machine is a server providing web, NFS and print services.
- Lethality = 5. This is a serious attack that, if successful will give the attacker full root privileges.
- System Countermeasures = 5. This is relatively secure system with all updated patches and running an IDS and is behind a restrictive firewall.
- Network Countermeasures = 3. The outer router should not have allowed inbound connections to port 515.
- Severity = (3 + 5)– (5 + 3) = 0. Very low risk.

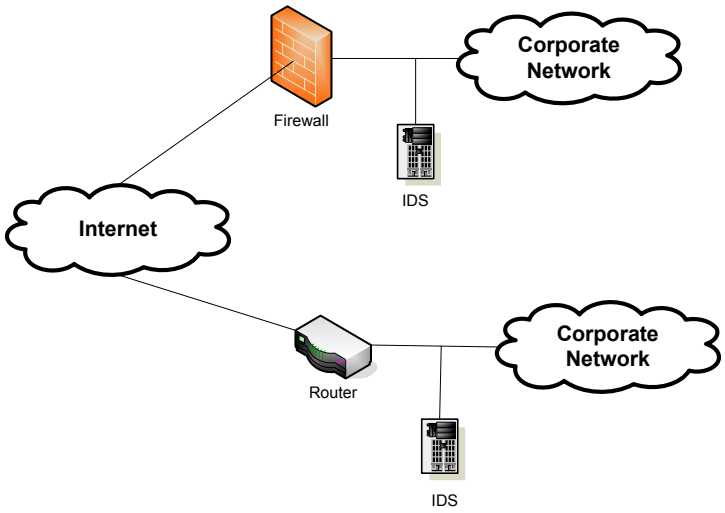
Defensive recommendations

- Though the attack did not succeed, this IP and/or the subnet it originates from must be blocked.
- Configure the outer firewall to block inbound port 515 connections. In addition, the source IP address must be blocked at the outer firewall.
- The entire subnet that this IP originates from has been blocked on the server firewall.
- The administrator for the source network has been notified and the relevant information has been sent over.

Implementing and Deploying an IDS

- Depending upon your network topology, an IDS may be deployed at one or more places depending upon what type of intrusion activities we wish to detect: internal, external or both.

- For example, if we want to detect external intrusion traffic only, and the network has only one router connecting to the Internet, the best place for an intrusion detection system may be just inside the router or a firewall.
- If the network has multiple paths to the Internet, we may want to place one IDS box at every entry point.
- However if we want to detect internal threats as well, you may want to place a box in every network segment.
- In many cases it is not necessary to have intrusion detection activity in all network segments and IDS placement may be limited only to sensitive network areas.
- Note that more IDS's mean more work and more maintenance costs. Ultimately the decision really depends on a company's security policy, which defines what resources must be protected from attackers.
- The diagram below shows typical locations where an IDS may be deployed.



- Typically an IDS should be deployed behind each firewall and router. In the case of networks that have demilitarized zones (DMZ), an IDS may be placed in that zone as well.

Security Zones and Levels of Trust

- A typical network design will segment networks into two broad areas, a secure (or private) area and an unsecured (or public) area.
- Networks are also divided into many different areas with each area having a different level of security policy and level of trust.
- For example, a company's finance and operations department may have a very high security level and may allow only a few services to operate in that area. In some cases no direct Internet services may be available from sensitive areas.
- However a DMZ or de-militarized zone part of a network may be open to the Internet world and may have a very different level of trust.
- Depending upon the level of trust and company security policy, different policies and rules for intrusion detection in different areas of your network must be implemented.
- Network segments with different security requirements and trust levels must be physically separated from each other.
- Install one IDS in each zone with different types of rules to detect suspicious network activity.
- As an example, if the finance department has no web server, any traffic going to port 80 in the network segment must be closely scrutinized for suspicious activity.
- However the same is not true in the DMZ zone where a there is a company web server accessible to the general public.

IDS Policy

- Before an IDS is deployed on a network, there must a clear and unambiguous policy in place to detect malicious traffic and take action when such activity has been identified and verified.

- The policy must clearly state the IDS rules and how they will be applied. The IDS policy should establish the rules for specific functions such as:

- **Monitoring the IDS**

    - Depending on the IDS, there will be alerting mechanisms that provide information about intrusion traffic.

    - These alerting systems may be in the form of simple text files, or they may be more complicated, perhaps integrated to centralized network management systems like a MySQL database.

    - The policy must clearly identify an analyst(s) with the requisite skills to monitor the intrusion activity.

    - The intrusion activity may also be monitored in real time using pop-up windows or web interfaces. In this case analysts must have knowledge of alerts and their meaning in terms of severity levels.

- **IDS Administration**

    - As with all systems, a skilled analyst must be assigned the routine maintenance of the IDS, including log rotation, archiving, etc.

- **Incident Handling**

    - Depending upon the severity of the incident, you may need to get some government agencies involved. The policy must clearly identify the incident response team and the escalation process (level 1, level 2 and so on).

    - The escalation process is basically an incident response strategy. The policy should clearly describe which incidents should be escalated to higher management and how.

- **Reporting**
    - Reports may be generated showing what happened during the last day, week or month.
    - Signature updates
    - Attackers and Hackers are continuously creating new types of attacks. These attacks are detected by the IDS only if it knows about the attack in the form of signatures.
    - Because of the continuously changing nature of attacks, the signatures must be updated together with the IDS rules.
    - In the case of an IDS such as Snort an analyst can update signatures directly from the Snort web site on a periodic basis or on his/her own when a new threat is discovered.

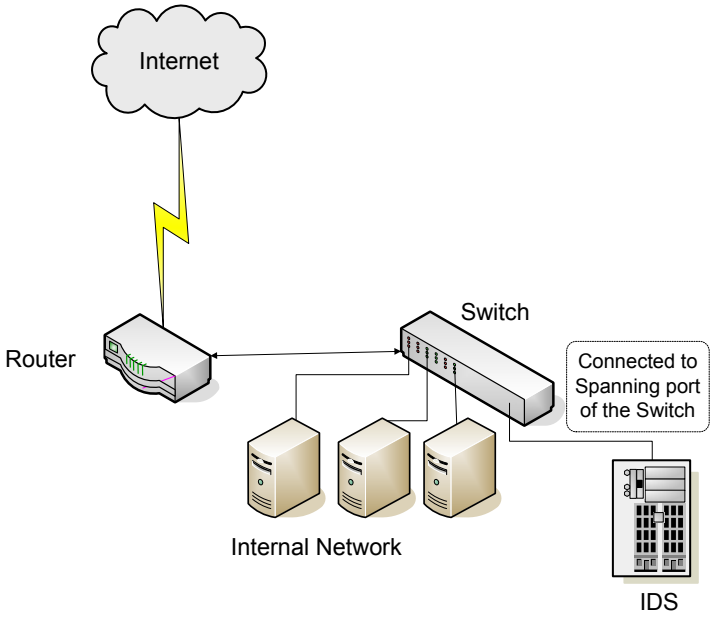
- **Documentation**
    - The IDS policy should describe what type of documentation will be done when attacks are detected.
    - The documentation may include a simple log or a detailed record of complete intrusion activity.

Monitoring Switched Networks

- Implementing an IDS in a switched environment presents several challenges due to the basic differences between standard hubs and switches.
- Hubs will echo every packet to every port on the hub, excluding only the port the packet came in on.
- A switch on the other hand is capable of inspecting the data packets as they are received, determines the source and destination device of that packet, and forwards that packet appropriately.
- By delivering messages only to the connected device that it was intended for, network switches conserve network bandwidth and offer generally better performance than hubs.
- However, an IDS connected to a switch will only receive packets that are transmitted on its port, defeating the purpose of an IDS, which to examine every packet entering and leaving the network.
- Therefore, deploying an IDS in a switched environment requires specific workarounds so that the sensor is able to see all of the network traffic.
- There are three main methods of inserting an IDS into a switched environment: **TAPS**, **Hubs** and **Spanning ports**.

Span Port

- A SPAN (Switch Port ANalyzer) port is available on high-end switches and it will configure the switch to copy the transmitted and received packets from one port or VLAN to another port, one which is connected to an IDS.
- Switches also only allow one port to be spanned at a time, so monitoring multiple machines can be difficult or impossible.
- In the diagram below the switch is configured to span the traffic from the port that the router is attached to, to the port the IDS is monitoring.



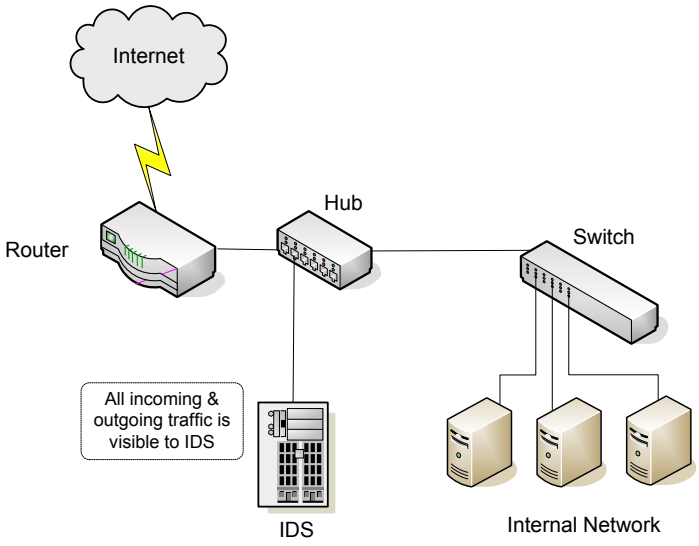
IDS connected a spanning port

- The advantages of this solution are:
    - Ease of installation. The IDS can be inserted into the network without modifying the core network architecture.
    - Management of the IDS does not require any additional hardware or special configuration.
    - Firewall configuration is not necessary to accommodate the IDS.
- The disadvantages of this solution are:
    - There is only one span port per switch. Monitoring more than one port will require the spanning of a range of ports.

- Spanning multiple ports can result in overloading the switch and degrading its performance.

Hubs

- The hub is placed between the connection to be monitored. This is usually between either two switches, a router and switch, or a server and switch, etc.
- The diagram below shows a hub between the router and the switch. This allows traffic to still flow between the switch and the router while the hub makes all the traffic visible to the IDS.



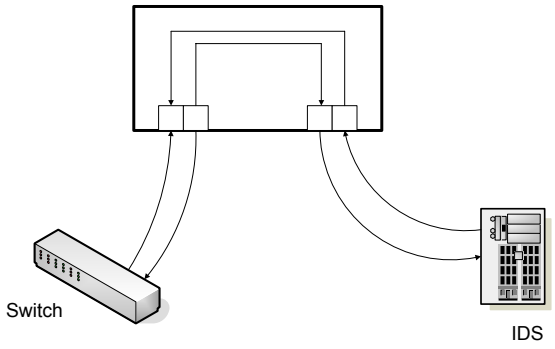
Connecting an IDS Using a Hub in a Switched Environment

- The advantages of this solution are:
  - Ease of installation and configuration. The IDS can be inserted into the network without modifying the core network architecture.
  - Management of the IDS does not require any additional hardware or special configuration.
  - Firewall configuration is not necessary to accommodate the IDS.
  - Will not require expensive switches with spanning ports.
  - Hubs are very cheap.

- The disadvantages of this solution are:
    - There will a significant degradation in throughput for full-duplex connections between the switch and router due to collisions.
    - Low cost hubs tend to be prone to failure. Fault tolerant hubs can be used but they are relatively expensive.

TAPS

- A tap is a network device that is used to create permanent access ports for passive monitoring. A tap can be set up between any two network devices, such as switches, routers and firewalls.
- Taps are fault-tolerant, single or multi-port full-duplex that maximize traffic visibility to an IDS and minimize link downtime on switched LANs.
- Taps provide access to all network traffic from both sides of a full-duplex link at the line data rate. They are completely invisible and non-intrusive to the network. .
- Taps are by design fault-tolerant; the main connection between the two devices it is connected to is hardwired.
- The following diagram illustrates how a single-port tap functions:



TAP Connections

- Note that a tap is unidirectional and passes traffic one way only. Thus, two connections are required between the switch and the tap.
- Typically only one connection from the tap to the IDS will suffice because that will prevent traffic to the IDS only and prevent two-way traffic between the IDS and tap.

- However, if the IDS is required to perform an active response such as terminating connections it will require two-way communication with the tap.

- Taps come in several configurations with the only difference between the configurations being the number of connections that the tap can monitor.

- Taps can also come in 4, 8, and 12 port versions. A typical tap will provide the following functionality:

- **Fail-closed**

    - The taps have circuitry that closes the connection when power fails, maintaining the network link.

- **Fail-open**

    - Where it is essential that all traffic is monitored some organizations prefer the taps to fail-open, this is more usual for Intrusion Prevention Systems than Intrusion Detection Systems.

- **Traffic Aggregation**

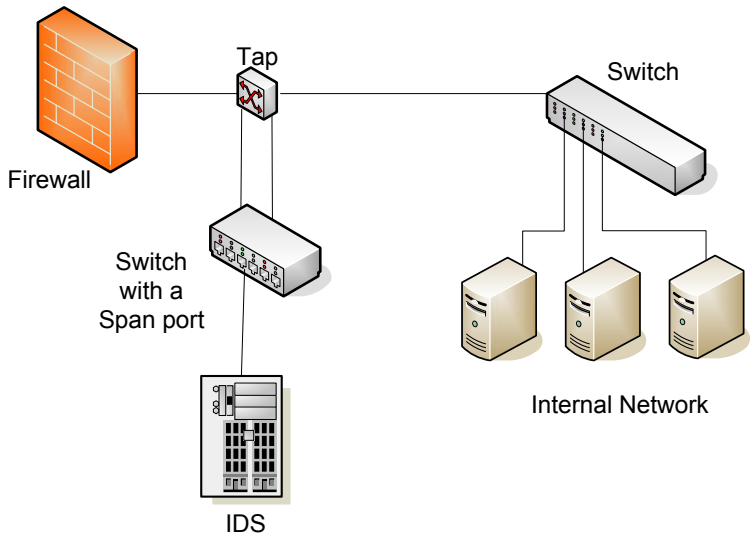
    - Taps can aggregate full-duplex traffic into a single stream so that a stateful IDS can see both sides of the conversation and therefore provide the lowest incidence of false positives.

    - Some Taps present dual outputs to combine the traffic externally, others combine the traffic internally presenting a single full duplex output.

- **Reset Injection**:

    - Some taps allow the IDS to transparently inject TCP **Resets** (**RST**) into the stream, terminating malicious sessions. This is referred to as "**Terminate Sessions**".

- Consider the next diagram that illustrates the use of a tap-based IDS on one side of a firewall.



Connecting an IDS Using a Tap in a Switched Environment

- The switch used to connect the IDS to the tap is configured so that all of the incoming ports are part of a single VLAN. The VLAN is then spanned to the IDS.
- Note that a hub could have been used instead of a switch with a spanning port but that will result in the previous disadvantages of fault-tolerance and throughput degradation.

- Current designs favor the use of taps in deploying IDS solutions. As always there are pros and cons to this approach also.
- The advantages of a Tapped solution are:
    - The tap is fault tolerant. Even if the power fails the connection between the router and the switch is hardwired in and requires no power to function.
    - A tap will not degrade the traffic.
    - Once the tap is in place, changes to the IDS will not impact the overall network architecture.
    - A tap will prevent attackers from establishing connections to the IDS host. In fact the IDS will be invisible to the outside world. This is also referred to as “stealth mode”.

- The disadvantages of a Tapped solution are:
    - Taps can be very expensive.
    - Terminate sessions capability will require extra configuration and additional hardware.
    - Monitoring two-way traffic will also require extra configuration and additional hardware.

Protecting the IDS Host

- A very important NIDS design issue protecting the system on which the intrusion detection software is running so it is not compromised.
- The first step to securing the sensor is to disable all network services on it.
- The most common method of exploiting a system is through its network services.
- The IDS sensor must be updated and patched with the latest releases from the vendor.
- Configure the IDS machine so that it does not respond to ping (ICMP Echo) packets.
- If the IDS is running on a Linux machine, use Netfilter/iptables to block any undesired traffic. The IDS itself will still be able to see all of the packets.
- The sensor must be purpose of intrusion detection and not for any other services. Disable all user accounts except the main root account to be used for managing the IDS.
- An IDS such as Snort can run on a stealth interface which only listens to the incoming traffic but does not send any data packets out.
- One way to accomplish this is to use a special cable on the stealth interface. Basically the cable shorts pins 1 and 2.
- Pins 3 and 6 are connected to same pins on the other side. More details on this can be found at http://www.snort.org/docs/faq.html
- We can also run the IDS on an interface where no IP address is assigned. For example, on a Linux machine, we can activate interface eth0 using the command "***ifconfig eth0 up***" without assigning an actual IP address.
- The advantage to this technique is that if the IDS host does not have an IP address, no other host can access it.
- If the sensor has a second interface card we can configure it so that it can be used to access the sensor itself.