

“My IDS/IDP Works Perfectly!” Famous Last Words or Informed Assessment?

By Philip Joung

philip.joung@spirentcom.com

Intrusion detection and prevention systems continue to gain market visibility and acceptance as an important part of a network's security infrastructure. Indeed, the continuing proliferation of network attacks has heightened the importance of these products. The latest trend is for such equipment to sit in the critical path of the network infrastructure, inspecting network traffic and blocking network access to rogue network attacks and intrusion attempts. This, however, imposes a serious responsibility on the intrusion prevention system, as it has the very important role of minimizing falses, whether positive or negative. Testing becomes a critical element in validating the security infrastructure, as a failed security network can have drastic consequences ranging from the merely annoying to potential financial and even criminal liability. Read on to discover ideas to effectively test the network's security and increase performance.

Ebbing the Flow of Network Attacks

Network security long ago lost its status as an optional part of a robust network infrastructure. Today, security must pervade the network in order for it to maintain nonstop functionality, reliability and viability. Security concerns go well beyond the scope of your internal networks, requiring agreements with business partners, SLAs with service providers, and protection from malicious external attacks.

While intrusion detection systems (IDS) have maintained an ongoing presence in the security scene for several years now, intrusion prevention systems (IPS) are just starting to mature and hence have enjoyed a recent surge in popularity. This article will briefly introduce IDSs and IPSs, which can be supplemented by other sources (books, Web sites, magazine articles, etc.) that contain much more detailed information on these technologies. You should instead expect to learn some of the issues with deploying these security devices and learn tips on effectively validating their performance, security and reliability.

Comparing IDS and IPS

While both types of systems must examine all incoming and outgoing network traffic, an IDS creates alerts when it detects malicious or intrusion traffic or behavior. These IDS alerts can be used by security administrators to locate and mitigate network security issues and serve as a record during network forensic investigations. IPSs also include a robust attack/intrusion detection engine but can also take the extra step of stopping the attack itself.

Networks often have their IDSs connected using a regenerative tap, which replicates *all* network traffic on behalf of the IDS. This ensures the IDS can inspect each and every bit that traverses the network. This connection topology means that most IDSs operate from the side of the

network. However, an IPS goes one step further, usually attached in the path of the network traffic just like a firewall, blocking detected attacks and intrusion attempts.

An effective IDS and IPS must minimize falses, with the most effective approaching 0 percent:

- ▲ False negative: Any attack or intrusion attempt that goes undetected by the IDS/IPS is a false negative. In this case, the undetected attack can successfully continue on its destructive path.
- ▲ False positive: In a healthy network, most activity will be normal, legitimate traffic. As a result, an effective IDS/IPS will not affect this traffic. This means not only minimizing the times that legitimate traffic is marked as an attack and blocked, but having little or no effect on performance and latency and not adversely altering traffic.

Because of differences in the way networks connect and employ IDSs and IPSs, the criticality of their performance and uptime may differ as well. Although failures in either device are problematic, an IPS failure can cause significant network outages directly attributable to the device itself. If an IPS fails open, attacks may breach the network without anything to stop them. Alternatively, an IPS that fails closed does certainly maintain a high level of security—of course, no other traffic passes, either. In such a case, a fail-close IPS becomes a successful denial-of-service attack. Does this mean that deploying and maintaining an IDS should be easier and take less effort? While this may sometimes be true, do not fall into this trap. One may as well not even deploy the IDS if efforts cannot be taken to maintain and ensure its nonstop operation, functionality and performance. IDSs and IPSs left to languish on their own will quickly become obsolete and ineffective.

Overall, securing a network isn't easy. Security administrators must master many unrelated skills while deploying and enforcing a system that undoubtedly adds hassles and inconveniences and reduces network performance. Policies must be drafted and enforced, budgets created and met, security devices deployed and maintained, servers secured and updated, physical access restricted and hardened, and overall security tested and audited. IDSs and IPSs have an important role within this security macrocosm.

Having confidence in their performance, security and reliability helps drive the overall quality of the network security infrastructure. What makes this possible? Testing.

Why Test?

Testing provides answers to many issues and questions that often cannot come from any other effort: How does my security infrastructure affect

overall performance? Does the network maintain acceptable performance while under attack? Do my redundant IPSs properly fail-over, and what are the effects on traffic during and after failover? At what future level of traffic will my systems need an upgrade? Testing discovers and helps resolve issues before network users are hampered by them.

Why expend all this extra effort to test the network? Among the arguments for skipping this step (and reasons to dismiss the argument) include the following:

- ▲ *It takes too much time.* There is no question that ramping up a testing effort takes time. However, once the initial startup takes place, a well-integrated testing strategy will drive overall network quality. Network, security and testing architects can collaborate from the beginning. This fosters a mindset of high quality and performance early on, creating a better network even though network deployment and testing times are reduced.
- ▲ *It's too expensive.* Studies have shown that networks undergoing testing have fewer performance and reliability surprises. The mitigation of one outage with testing pays for itself many times over. One can also amortize the testing startup costs over many years, as the investment in testing equipment and expertise will persist for many network iterations. Think of testing as low-cost insurance for your network.
- ▲ *Our vendor has already proven performance in their own tests.* Vendors who care about their products will expend their own efforts to validate the products they sell. Networks, however, vary dramatically in each deployment, and vendors do not have the ability to test their products in all situations and conditions. While they care about how their products perform, when a network fails as a result of their product, they may lose you as a customer, but you may lose your business. Which is the greater liability? Vendors often publish performance numbers based on best-case scenarios. Competitive pressures and marketing demands make this commonplace.
- ▲ *Our network already works quite well.* This is often the hallmark of good network design and practices. However, as the network ages, its performance will change, and testing can help determine the implication of these changes. Examples of changes that affect performance: increasing number of users, a growing database, an IPS with an increasingly complex detection set, and file servers storing hundreds of gigabytes or terabytes.

Few networks can tolerate unexpected outages. Most office workers are familiar with the dramatic drop in productivity during LAN outages. Consider a network outage at a large supermarket with its cash register systems down, or a 911 emergency call center with its dispatch systems out. At best, outages are simply annoying, but downtime of the wrong system at the wrong time can cause significant revenue loss and even potential loss of life. Testing is not a panacea, but it can be extremely powerful insurance against outages.

Testing an IDS or IPS

Consider the most important characteristics in a network device. For many security administrators, the most obvious is security. Reliability, availability, scalability and performance form the remainder of what many consider the minimum requirements for a robust network. Each of these characteristics has a unique methodology to test it—the following will help get you started.

Limitation	What increases this?	Testing tips
Memory	Many concurrent users, slow connections, large packet sizes	<ul style="list-style-type: none"> • Use tests with long user sessions and long user think times • Test using slow network connection speeds, which take more memory because the memory must retain incoming and outgoing data until the transfer completes • Most IDS/IPSs must cache packets before sending to its inspection engine, so larger packets tend to use more memory
CPU	High transaction rate, high alerts, small packet sizes	<ul style="list-style-type: none"> • High transaction rates require more CPU resources • Tests that contain many transactions that the IDS/IPS will alert increased CPU utilization • Small packet sizes tend to reduce memory consumption and increased CPU utilization
Bandwidth	Large file transfers, high transaction rate, DDoS attacks	<ul style="list-style-type: none"> • Large file transfers tie up bandwidth, especially after it passes inspection • Many small transactions can add up to high bandwidth utilization if the IDS/IPS can keep up • DDoS attacks by their very nature utilize bandwidth

Figure 1: The three major limitations of most network devices, including intrusion detection and prevention systems, and ideas on testing each limitation

Factor	Tools	Testing tips
Security	Attack simulator, Load testing tool	<ul style="list-style-type: none"> • Send various attacks, both alone and combined with other attacks • Send attacks combined with normal traffic • Test at different levels of normal traffic and hide one attack among the "haystack" of normal traffic
Reliability	Load testing tool with robust reporting	<ul style="list-style-type: none"> • Send varying levels of normal traffic, checking that falses do not occur • Send traffic and ensure that traffic does not fail (timeouts, bad sequence numbers, bad headers, etc.)
Availability	Load testing tool	<ul style="list-style-type: none"> • Send large amounts of traffic over several days (or weeks) to see if failures occur over time • Connect two IPSs in a high-availability configuration, start traffic, induce failure in one (for example, pull the network plug) and determine the amount of transparency in the failover
Scalability	High-capacity load testing tool	<ul style="list-style-type: none"> • Predict network traffic levels over the next several years and test that level of traffic • Confirm that the vendor's IDS/IPS scalability strategy works by asking the vendor to loan the extra equipment and test it
Performance	High-capacity load testing tool	<ul style="list-style-type: none"> • Test for concurrent user ability • Test for system transaction rate • Test for average transaction latency at different concurrent user loads

Figure 2: Tailoring your testing for different factors require distinct methodologies. This table contains some tips on making this testing more effective

Network Introspection and Characterization

Each network has many factors unique to it, including its topology, its purpose(s), its users, and the traffic they generate. As a result, testing should start with an assessment of these factors, looking at the ones that contribute most to the characteristic(s) being tested.

User traffic and behavior is an obvious one. Quantifying users involves capturing their particular behaviors, which can include their network usage patterns, application versions, delays, and user abandonment. For example, a network supporting a Web site should quantify things such as typical site usage patterns, browser versions most often used, delays between page clicks (often called think time), and the incidence of "click-aways," where the user gets frustrated and leaves. For IDSs and IPSs, attacks and intrusion attempts will be among the traffic patterns tested.

Networks also have certain issues that dramatically affect performance and stability. This becomes more and more likely with larger and more

complex networks, with the Internet being a prime example. Issues faced by networks include end-to-end latencies, link speeds, packet loss, IP fragmentation, jitter, and bursty traffic patterns. Any of these factors, if severe enough, dramatically affect performance.

Finally, the topology and setup of the network will affect testing. The closer the testing network replicates conditions, topology and settings seen (or expected) in production, the higher the confidence in the testing results.

How much realism and characterization is needed for testing? This answer ends up being a judgment call on the tester's part, balancing the effort and expense of incorporating increased realism with the costs and criticality of failure in the network. However, prioritization will help, choosing user and network behaviors with the most dramatic effect on your network.

Choose Your Weapon

Many tools make testing faster, more reliable, and more effective. In fact, testing tools are a necessity for most modern networks. Manual testing can no longer reach the performance levels needed to determine the network's limits, nor can it support the repeatability required for ascertaining the performance changes from network updates. As with anything added to your network, several factors contribute towards selecting the right testing tool(s):

- ▲ Price: Consider the overall costs of using the tool, including acquisition costs of hardware and software, training, support, maintenance and customization.
- ▲ Performance: The most useful tools will exceed the expected network traffic levels, hopefully by a wide margin. For IDS/IPs, having a tool that exceeds the bandwidth and transaction rate ensures complete assessment of the IDS/IPS for your network.
- ▲ Ease of use: Frequent use of a testing tool maintains network performance, often driven by how easy the tool is to use.
- ▲ Configurability: A tool that maximizes control of the traffic it generates enables more realistic, robust testing. Look for tools to support configurable loads, configurable client IP subnets and IP addresses, and scripting support.
- ▲ Appropriateness: Test tools abound, filling testing needs for particular users and particular testing niches. Ensure the tool matches (or has the flexibility) to match the required testing. For example, the tool should support the protocols most prevalent and most important to your network, such as HTTP, FTP, SMTP, SSL, and streaming protocols.
- ▲ Security attacks: One of the primary goals of IDSs and IPSs is the detection and termination of attacks. Tools that simulate these attacks should be included in your test arsenal.
- ▲ Realism support: This incorporates user and network realism as mentioned above in the Network Introspection section.
- ▲ Standardization: A tool widely used by the industry will not only ensure the longevity of your testing investment, but increase the chances of attaining support when needed. Look to colleagues and the experts (test labs, magazines) for referrals on tools to use.
- ▲ Reporting: having a tool that generates excellent test traffic and getting everything right and realistic is useless without reporting. Strong reporting capabilities provide details on what happened during the test and how to mitigate issues.

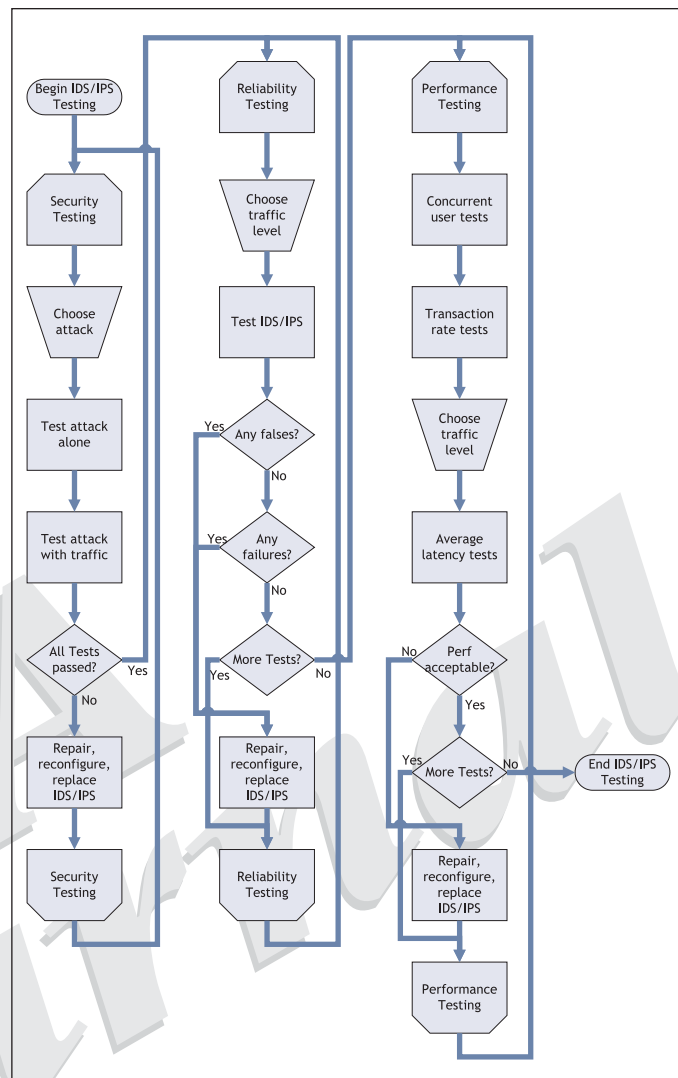


Figure 3: An example of the steps required to test an IDS or IPS

Bombs Away!

Now that the network traffic is characterized, and the tools are available to support this testing, what's next? Testing, of course!

An effective testing strategy will methodically go through several steps to assess the network or device under test. Keep in mind that the steps described later tests an IDS/IPS using representations of your traffic determined earlier and within the context of your network. As a result, it differs from other methodologies for testing the IDS alone. This focused testing has value, resulting in the performance numbers published in vendor datasheets or published in magazine reviews. Use these numbers as the best-case scenario, as your own traffic will almost certainly produce lower results.

With all the effort expended earlier into characterizing your network's traffic, why not use that traffic in one test to assess the IDS/IPS? Because network traffic evolves over time, a string of tests helps assess overall behavior of the IDS/IP under varying conditions. This forms a more complete picture to reduce the likelihood of surprises in production deployments. Multiple tests also mask the inaccuracies and assumptions that occur in almost all traffic characterizations.

Every IDS and IPS has limitations based on its combination of software and hardware. Although several limitations exist, the most important are memory, CPU and bandwidth. Any test will exercise all of these to a cer-

tain extent, but tailored tests can target specific limitations to help determine the maximum potential of the IDS/IPS. See Figure 1 for tips on testing these limitations.


Next, determine the most important factors to stress in testing: security, reliability, availability, scalability, and performance. For IDS/IPS, this list will typically be at least security, reliability and performance. Consult Figure 2 for ideas on how to test each of these factors.

With the many steps required to test an IDS properly, a flowchart can help make the process easier to visualize and follow. Figure 3 shows an example of the steps towards assessing an IDS or IPS. These steps should be tailored for your network's priorities and requirements.

IDSs and IPSs are complex systems having a lot of responsibility for securing a network. Testing the systems should not be taken lightly, but with a little planning, setup and design, one can create a solid network with high security and excellent performance.

Teaming up for Successful Networking

Network security is not easy. You should strive to collaborate with others to improve the state of network security, as going solo is rarely successful. That is how deployment and testing of the security network infrastructure should be—teamwork among different individuals, departments and contributors. Each person or group brings new perspective and value to the effort.

This article has provided a quick introduction into some of the ideas, approaches and issues in IDS/IPS testing, but there is much more to a complete assessment of these devices. Your investment in robust testing will ensure you and your organization reliably gain the security advantages from IDSs and IPSs. This peace of mind is priceless. 

Philip Joung has more than 17 years of IT experience and is currently the director of technical marketing at Spirent Communications.