ixia

Denial of Service (DOS) Testing

IxChariot

Contents

Ixia
26601 W. Agoura Road
Calabasas, CA 91302
Phone: (818) 871-1800
Fax: (818) 871-1805
Email: info@ixiacom.com
Internet: www.ixiacom.com

Denial of Service (DOS) Testing: Sample Test Plans

Denial of Service (DoS) attacks are a reality for most organizations with connections to the public Internet. In order to protect yourselves from the potential hazards of network hackers and malicious coders, a set of devices and software-based tools such as DUTs, intrusion detection systems (IDS), remote access solutions (VPN) and sophisticated routers and L4-7 application switches have been developed to effectively block malicious traffic and protect the organization's data and information infrastructure.

Leveraging the advanced functionality of Ixia hardware, IxChariot is now capable of generating line-rate traffic that emulates common DoS attack types while at the same time generating and measuring the performance of application traffic (VoIP, Internet, enterprise) that is being sent over the network.

This test primer outlines how IxChariot software can be used on an Ixia platform to:

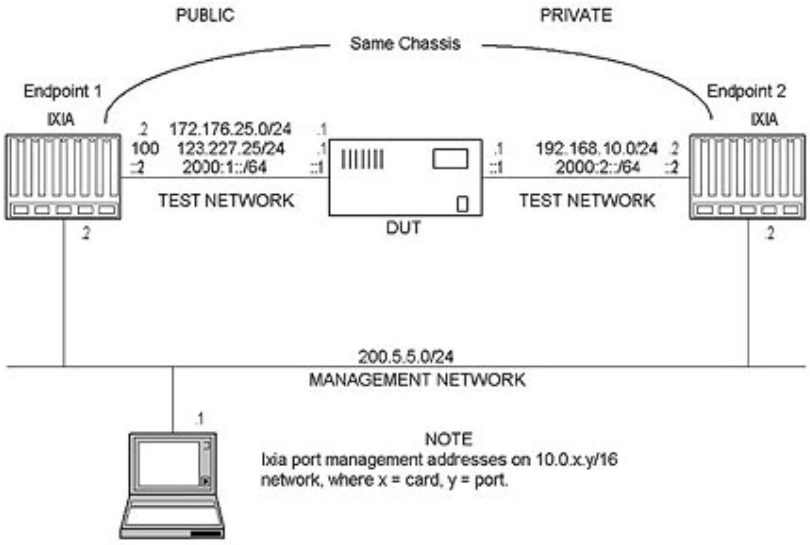
- assess the impact of DoS traffic on existing network traffic
- measure the performance of devices responsible for denying DoS traffic network access
- determine the performance of network devices that are being attacked (e.g. servers, routers, WLAN access points)

Overview of Denial of Service functionality in IxChariot

T he test cases for IxChariot's integrated DoS/application traffic emulation can be split into two principal groups:

- generating DoS/application traffic between two Ixia ports and measuring the performance of the overall network
- directing DoS traffic against a specific device (e.g. firewall) while generating and measuring the performance of application traffic being sent between two IxChariot Performance Endpoints

In each case, performance measurements need to be taken with both DoS filtering/ blocking solutions being enabled as well as disabled.



A brief outline of the DoS attack types supported in IxChariot

IxChariot has the ability to produce several common types of DoS attacks, as explained below. Keep in mind that such attacks are created within Ixia-specific hardware, generated directly using Field Programmable Gate Arrays (FPGA), and as such, this malicious traffic can be generated in speeds ranging from zero to the full wire speed of the interface.

SYN Attack
Every TCP connection begins with a single TCP SYN flag being sent from the client host to a server. In response to receiving such a flag, the server typically allocates resources and then sends a TCP SYN-ACK packet back toward the client host station. A SYN attack overwhelms the victim computer with a rapid succession of SYN packets, causing it to over allocate resources and either crash or wait for the allocated resources to time out.

Teardrop Attack
Fragmented packets that continuously overlapping offsets are sent from a client to a server. The server cannot reconstruct the original payload from the fragmented overlapping packets and eventually crashes.

Ping Attack
An ICMP Ping Request is sent to a server at a high rate, causing bandwidth problems on the server's network.

Ping of Death (POD) Attack
ICMP Ping Requests are sent from a client to a server; however, each packet is a fragment of a complete Ping Request of extremely large size. This may cause the server to over allocate resources and crash.

Unreachable Host Attack
An "ICMP Host Unreachable" message may be sent to a server that is already in communication with another host. This will likely cause the server to drop that connection. Test case examples

Three test cases are observed in the following sections. These simple test cases show how to set up IxChariot for testing the performance of networks and specific devices when being loaded with both DoS and standard application traffic.

Figure 1 shows the setup that will be used for the test cases outlined below. All addresses used in the course of testing appear in this Figure, including both IPv4 and IPv6 addresses. Only two physical Ixia ports are used in this scenario.

Test Case 1: Ping Attack on Oracle Traffic

Objective

In this test case, you will create a typical traffic pattern of Oracle traffic, operating over a known port. After measuring throughput, response time and transaction rates, you will start a DoS Ping attack and compare the results.

Methodology

1. Create an Oracle transaction as per the following criteria:
   - Endpoint 1 and Endpoint 2 network addresses assigned as per your setup
   - Set up the "How does the Console know Endpoint 1" address using the Ixia management address, e.g.; 10.0.3.1. Similarly, set up a management address for the "How does Endpoint 1 know Endpoint 2?" address, e.g.; 10.0.3.2
   - Use a sample script such as Oracle_AP_Tier2_Invoice_Mult_Dist.scr" script for this pair.
   - Edit the script so that it uses TCP port 2321. Also, use minimum transaction delays and unlimited send data rate.
   - Replicate pairs n times according to your traffic volume. In this case, the pair was replicated 19 times so that there are a total of 20 Oracle traffic flows in this test.
   - Set the run time for 1 minute. Be sure to choose the 'Batch mode' run option.
   - In the resulting charts, you may want to display a total rather than the results of each individual script.

2. Ensure that the DUT allows all traffic (rules not enforced) to pass through to generate a set of baseline test results.

3. Run the transaction and record results in the "No DoS/No DUT" column in table 1 below. See Figure 2 for a typical results screen.

4. Now instruct the DUT to enforce its rules, and then re-run the transaction. Record the results in the "No DoS/DUT" column in table 1 below. This should tell you whether or not the DUT is affecting traffic rates. Note that the DUT should be set up to defend against the DoS Ping attack, which will be used in a subsequent step.

5. Now instruct the DUT to NOT enforce its rules and insert a DoS attack. This particular attack should emanate from Endpoint 1 and target Endpoint 2.

6. Depending on the capabilities of your DUT and the available network bandwidth, you may want to override the stream line rate (e.g. 15%). Also, ensure that the "Measure hardware performance pair statistics" checkbox is in the UNCHECKED position. This will ensure that the target CPU (in this case, the Ixia CPU) handles the details of the incoming ICMP Ping requests. See Figure 3 for a description of what this should look like. Run and record the results in the "DoS No DUT" column of table 1. For a comparative look at results with DoS and no DUT, see Figure 4.

7. Now enable the DUT and re-run the transaction. Record the results in the "DoS / DUT" column in table 1 below.

8. The completed table should give you a good comparative analysis of how well your DUT can protect internal hosts from DoS Ping attacks.

Description	No DoS No DUT	No DoS DUT	DoS No DUT	DoS DUT
Throughput Average				
Throughput Minimum				
Throughput Maximum				
Throughput Average				
Throughput Minimum				
Throughput Maximum				
Throughput Average				
Throughput Minimum				
Throughput Maximum				

Table 1: Comparative test table for enterprise application traffic with DoS attack traffic.

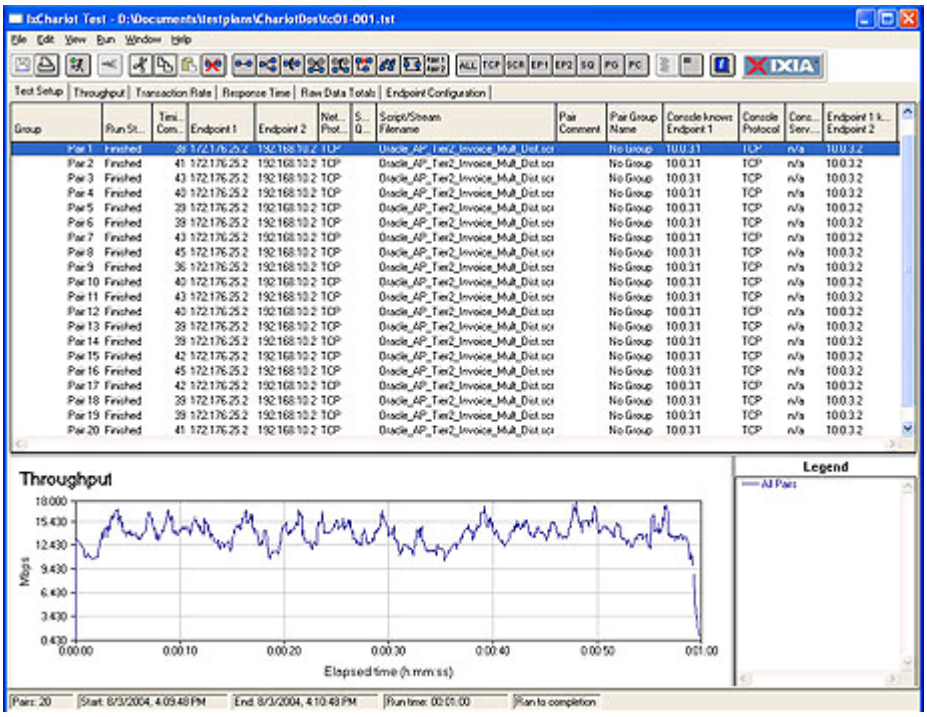


Figure 2: Typical results for no DoS and no DUT.

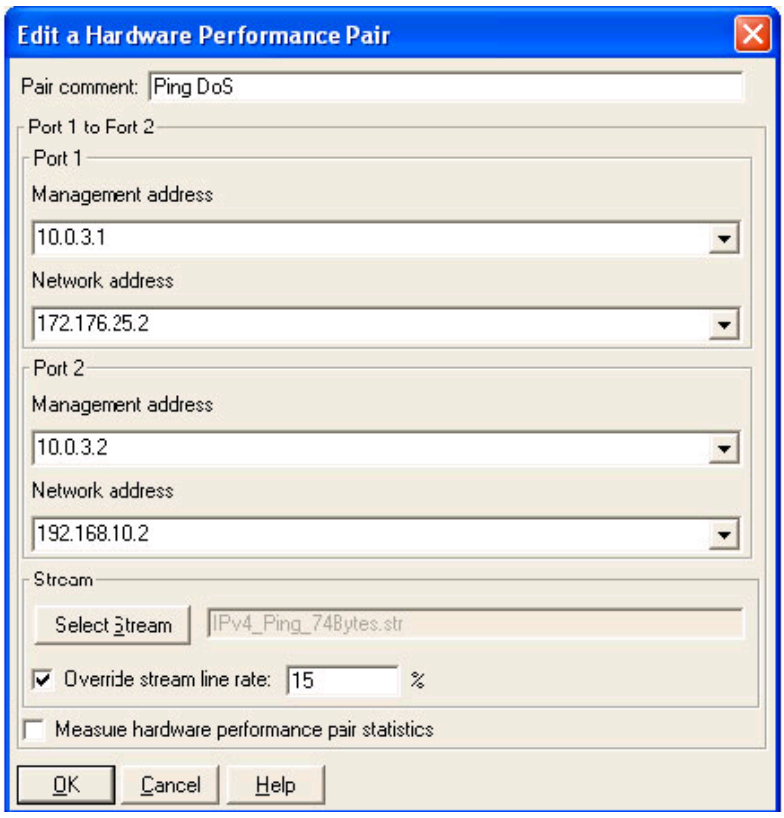


Figure 3: Setting up for DoS Ping attacks.

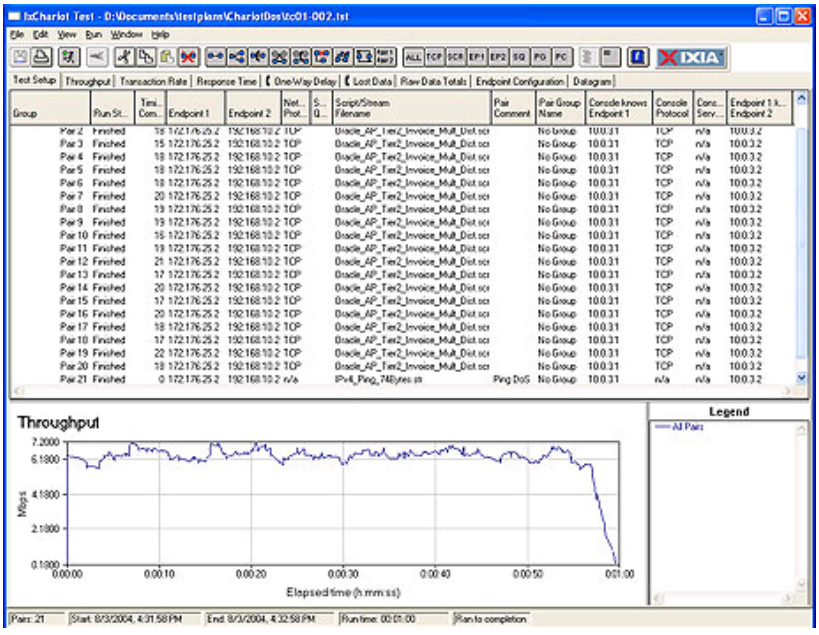


Figure 4: Comparative results for DoS and no DUT

Test Case 2: VoIP and TCP SYN Attacks

Objective

This test case will observe what happens in a VoIP environment when a series of SYN attacks are directed at a host. It will show how VoIP connections can be sabotaged when the DUT is not configured correctly.

Methodology

A communication channel will be set up between two sets of Performance Endpoints, creating several VoIP conversations. A TCP SYN attack will be launched against one of the endpoints from a third location. The DUT will initially allow the attack to proceed. In the second iteration, the DUT will be trained to disallow any TCP connection attempts from the third party location.

Figure 1 will be used to model the attack. The VoIP conversations will exist between the 172.176.25/24 and the 192.168.10/24 networks. Another network will be superimposed on the same physical connection as the 172.176.25/24 network, utilizing an address from the 123.227.25/24 network.

Note that IxApplifier will be used to install an address range of 172.176.25.101 to 172.176.25.120 on the public side of the DUT, and address range of 192.168.10.101 to 192.168.10.120 on the private side. Also, the third party address of 123.127.25.100 will be installed on the external port. It will target address 192.168.10.101 on the internal port, using the DUT as a gateway at address 123.227.25.1.

1. Set up VoIP pairs between internal and external clients. The VoIP traffic will travel in both directions; that is, half of the connections will have Endpoint 1 on the private side of the DUT, and half will have Endpoint 1 on the public side. Set up a total of 20 pairs. See the setup in Figure 5.

2. Ensure that the DUT rules enforcement is turned off.

3. Set up the attacking port to use a hardware performance pair. Select the IPv4_Syn_Port80_74Bytes.str pattern. Make sure “Measure hardware performance pair statistics” is left unchecked so that the victim port (internal port) responds to the attack. Finally, override the stream line rate and set up a very low rate of attack. You may have to experiment a bit with this number. A good starting point is 0.01%. The objective is to slowly overwhelm the victim port as the test is underway.

4. Set the run time for 1 minute, and run the test. You should see the MOS scores registering an almost perfect performance up to the point that the victim processor gets overwhelmed with TCP requests. At that point, the MOS scores will cease to exist, indicating that no more data is coming in from the victim port.

5. If the target port did not crash, go back to step 3 and adjust the stream line rate higher. You should not have to go any higher than 5% to see the detrimental results of a TCP SYN attack.

6. Turn on DUT rule enforcement. There are several things that can be done to the DUT, depending on the sophistication of the filtering required. If the DUT terminates and proxies TCP connections, then you could turn on TCP SYN-Cookies to stop the effects of the SYN attack. The simplest method is to simply filter on the malicious address, 123.227.25.100. This may not be practical in the real world, but it can demonstrate the DUT’s ability to filter on undesirable source addresses.

7. Run the test again and ensure that the MOS scores stay at their near-perfect levels throughout the test.

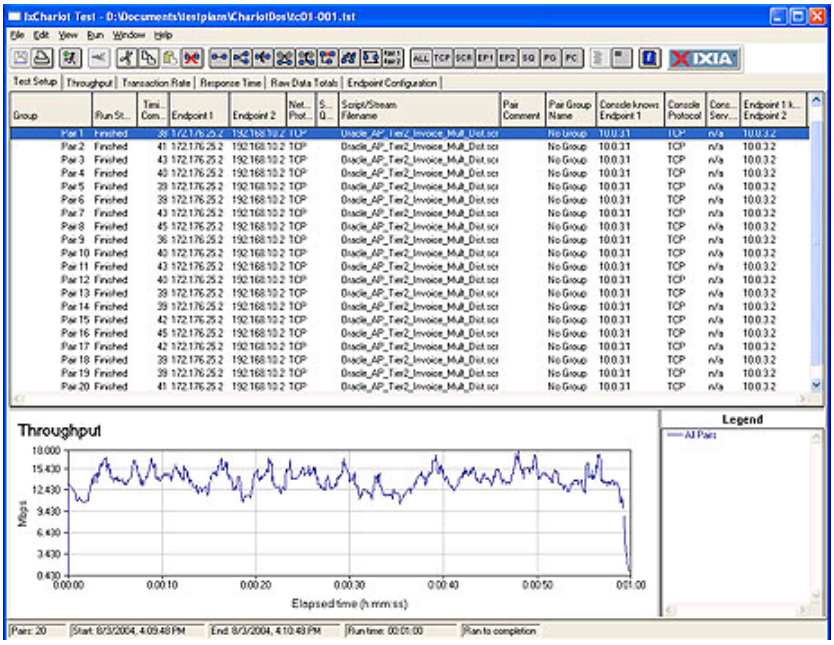


Figure 5: VoIP setup