The Science of Industrial Cyber Security

DATA-DRIVEN SOLUTIONS TO PROTECT CRITICAL INFRASTRUCTURE

ACHILLES LEVEL 1 PUBLIC REPORT

DeltaV MD Series 2 with DeltaV Firewall

401 W GEORGIA STREET, SUITE 1680 | VANCOUVER BC CANADA | V6B 5A1 | TEL +1 604 669 6674 | INFO@WURLDTECH.COM

Disclaimer

Wurldtech Security Inc. retains the right to change information in this report without notice.

Wurldtech Security Inc. believes the information in this report to be reliable and accurate but it is not guaranteed. Using and relying on this report is at your sole risk. Wurldtech Security Inc. is neither liable nor responsible for any loss, damage or expense arising from any omission or error in this report.

WURLDTECH SECURITY INC. GIVES NO WARRANTIES, EXPRESS OR IMPLIED. ALL IMPLIED WARRANTIES, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT ARE DISCLAIMED AND EXCLUDED BY WURLDTECH SECURITY INC. IN NO EVENT SHALL WURLDTECH SECURITY INC. BE LIABLE FOR ANY CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES, OR FOR ANY LOSS OF PROFIT, REVENUE, DATA, COMPUTER PROGRAMS, OR OTHER ASSETS, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

This report does not constitute a recommendation, endorsement or guarantee of any of the hardware or software products tested or the hardware and software used in testing the products.

The testing does not guarantee that there are no defects or errors in the products. It does not guarantee that the products will meet your requirements, expectations or specifications or that they will operate without interruption.

This report does not imply any sponsorship, endorsement, affiliation or verification by or with any company mentioned in the report.

All trademarks, service marks, and trade names used in this report are the trademarks, service marks, and trade names of their respective owners. No endorsement, sponsorship, affiliation or involvement in either the testing, the report or Wurldtech Security Inc. is implied nor should it be inferred.

Trademarks

Achilles™ is a registered trademark of Wurldtech Security Inc. in Canada and/or other countries.

Executive Summary

Wurldtech Security Technologies was engaged by Emerson Process Management to conduct cyber security testing of its DeltaV MD Series 2 with DeltaV Firewall against a variety of Ethernet and TCP/IP cyber attacks. The purpose of the testing was to determine if the device met the criteria for Achilles Level 1 Certification, the de facto industry standard for network robustness.

This document provides a summary of the test results and a description of the executed tests. Tests were performed by Wurldtech using the Achilles™ Satellite version 1.3. All testing was completed at Wurldtech Labs in Vancouver, BC, Canada from May 10, 2007 to May 15, 2007.

The test results indicate that the DeltaV MD Series 2 with DeltaV Firewall meets the Achilles Level 1 Certification conformance requirements as defined in *"Achilles Level 1 Conformance Requirements for Embedded Devices" on page 2*. The DeltaV MD Series 2 with DeltaV Firewall has earned Achilles Level 1 Certification.

The hardware revision of the DeltaV was 5.17 and the software version was 9.3.0. For the DeltaV Firewall the hardware revision was 000000DE and the software version was 3.1.1.0.emerson.

Achilles Level 1 Conformance Requirements for Embedded Devices

Pass/Fail Criteria

The Discrete Monitor and the ICMP Monitor are used to verify the operational continuity of the DUT during execution of the Level 1 test cases. The DUT must maintain an acceptable level of functionality during execution of each test case. To meet this requirement, the status of the Discrete Monitor and the ICMP Monitor must remain *Normal* during test execution provided that the test case is executed on or below 10 percent link utilization. For example, on a 100 Mbit link, all test cases are executed on or below 10 Mbits/s.

Exceptions to the above requirement are considered under circumstances in which all of the following hold true:

- The DUT behaves as designed.
- The DUT contradicts the Level 1 pass/fail criteria.
- An official description of the design is provided to Wurldtech.

For example, suppose the DUT is designed to process up to 200 packets per second on a 100 Mbit link and ignore all others. Such behavior contradicts the requirement to maintain visibility (ICMP Monitor) at link utilization up to 10 Mbits/s. When the DUT is tested, if Wurldtech can verify that the device is functioning as designed and the vendor can provide Wurldtech with the relevant design documentation, the pass criteria will be adjusted accordingly. Such adjustments are in line with Wurldtech's view that a key utility of certification is to ensure symmetric information, between the vendor and end user, regarding a device's network robustness and resilience.

Parameters for Level 1 Certification

- Max link utilization: 10 percent.
- Discrete Monitors: Cycle period = 1000 milliseconds. Tolerable period error = 4 percent.
- ICMP Monitors: Timeout = 0.5 seconds. Tolerable packet loss = 10 percent.

Test Cases for Level 1 Certification

The set of test cases required to pass Achilles Certification are listed in the test results summary on the next page. Note that this list of test cases is different from the current list of Achilles Level 1 test cases, as some test cases have been refactored, either by splitting one test case into multiple test cases, or combining multiple test cases into one test case. Despite the test reorganization, the set of all packets sent during an Achilles Certification has not changed.

DeltaV MD Series 2 Test Results Summary

Hardware Revision: 5.17
Firmware Revision: 9.3.0
Notes: The DeltaV passed certification with the assistance of the DeltaV Firewall and custom rulesets. The hardware version of the firewall was 000000DE and the software revision was 3.1.1.0.emerson.
Certification Level: CERTIFIED achilles LEVEL ONE

Family	Test Case	Result
Discovery	Open TCP Discovery	Pass
Discovery	Open UDP Discovery	Pass
Data Link	Random Frame Ethernet Storm	Pass
Data Link	Unicast Frame Ethernet Storm	Pass
Data Link	Broadcast Frame Ethernet Storm	Pass
Data Link	Multicast Frame Ethernet Storm	Pass
Data Link	Ethernet Fuzzer	Pass
Data Link	Ethernet Grammar	Pass
Data Link	ARP Flood	Pass
Data Link	ARP Grammar	Pass
Network	Random IP Packet Storm	Pass
Network	Unicast IP Packet Storm	Pass
Network	Broadcast IP Packet Storm	Pass
Network	Multicast IP Packet Storm	Pass
Network	Fragmented IP Packet Storm	Pass
Network	IP Fuzzer	Pass
Network	IP Grammar - Field Fuzzer	Pass
Network	IP Grammar - Fragmentation	Pass
Network	IP Grammar - Options Fields	Pass
Network	Destination Unreachable	Pass
Network	Type/Code Cross Product	Pass
Network	ICMP Grammar - Field Fuzzer	Pass
Network	ICMP Grammar - Echo Request	Pass
Network	ICMP Grammar - Echo Response	Pass
Network	ICMP Grammar - Parameter Problem	Pass
Network	ICMP Grammar - Source Quench	Pass
Network	ICMP Grammar - Redirect	Pass
Network	ICMP Grammar - Time Exceeded	Pass
Network	ICMP Grammar - Timestamp	Pass
Transport	TCP SYN Flood	Pass
Transport	TCP/IP LAND Attack	Pass
Transport	TCP Fuzzer	Pass
Transport	TCP Grammar - Field Fuzzer	Pass
Transport	TCP Grammar - Options Fields	Pass
Transport	TCP Grammar - Urgent Data	Pass
Transport	UDP Fuzzer	Pass
Transport	UDP Grammar	Pass

Appendix A — Brief Test Case Descriptions

A.1 Types of Test Cases

Test cases can be divided into several types.

A.1.1 Scans

Port scanning tools are often included in standard IT security procedures on IT and public networks. They are also one of the first tools an attacker uses to gather information about a system in order to exploit it. As more control systems become connected to networks or the Internet, it is important that the devices can withstand such scans.

A.1.2 Storms

In general, the ability of a device to handle packets varies with packet rate, so it is useful to execute each storm test case several times using different rate limits to determine the maximum values that the DUT can handle.

Rate limiting controls the number of test packets sent per second. When rate limiting is turned on, the test case sends packets at the specified rate. A higher rate limit means more packets are sent per second which increases the processing overhead for the DUT.

A.1.3 Grammars

Grammars define a domain of tests and provide coverage over that domain. There are two types of Achilles grammar:

- Grammars that describe and generate specific types of packets. For example, a grammar could generate packets where the IP length is always longer than the Ethernet length. The grammars generally generate invalid packets. In some cases, they generate and send valid packets to establish a specific protocol state with the DUT prior to sending the test packets.
- Grammars that generate invalid packets using a combination of header values taken from a predefined list.

Each grammar tests a specific protocol implementation or function of the protocol stack. If the protocol cannot handle invalid packets correctly, anomalous behavior may occur on the DUT.

A.1.4 Fuzzers

Fuzzers generate valid and invalid packets with randomized header values. They test a specific protocol implementation or function of the protocol stack. If the protocol cannot handle invalid packets correctly, anomalous behavior may occur on the DUT.

A.2 Discovery

A.2.1 Open TCP Discovery*

Open TCP Discovery performs the following:

- TCP port scans.
- TCP-level operating system and application fingerprinting.

A.2.2 Open UDP Discovery*

Open UDP Discovery performs the following:

- UDP port scans.
- Version scanning at the UDP level to differentiate between open ports and network stacks that do not conform exactly to the UDP protocol.

A.3 Ethernet

A.3.1 Random Frame Ethernet Storm*

Random Frame Ethernet Storm examines DUT communication over a deteriorating Ethernet link. The test case simulates a virus or malfunctioning device by generating large numbers of Ethernet frames and sending them over the Ethernet link.

As the destination MAC address is randomly generated, very few frames are addressed to the DUT. Subsequently, the DUT ignores most of the test case traffic and the network stack processing cost is low. The point of failure during this test is the Ethernet link.

A.3.2 Unicast Frame Ethernet Storm*

Unicast Frame Ethernet Storm generates unicast Ethernet frames and sends them to the DUT at a specific frame rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

Each frame contains a null payload. The source MAC address is set to the satellite MAC address and the destination MAC address is set to the DUT MAC address.

A.3.3 Broadcast Frame Ethernet Storm*

Broadcast Frame Ethernet Storm generates broadcast Ethernet frames and sends them to the DUT at a specific frame rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

Each frame contains a null payload. The source MAC address is set to the satellite MAC address and the destination MAC address is set to the broadcast address.

A.3.4 Multicast Frame Ethernet Storm*

Multicast Frame Ethernet Storm generates multicast Ethernet frames and sends them to the DUT at a specific frame rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

Each frame contains a null payload. The source MAC address is set to the satellite MAC address.

** Please contact Wurldtech for the complete test case description.*

A.3.5 Ethernet Fuzzer*

Ethernet Fuzzer generates invalid Ethernet frames with randomized protocol types and sends them to the DUT to examine the DUT's ability to maintain both view and control while dealing with the frames. Ethernet Fuzzer tests how the DUT handles unsupported protocols and how it handles invalid frames for supported protocols.

A.3.6 Ethernet Grammar*

Ethernet Grammar generates Ethernet frames with various Ethernet types, addresses and frame length values. It tests the interactions of these values with the IP addresses and length of the payload. The generated frames are invalid and the DUT should drop them. The grammar tests whether the DUT makes invalid assumptions about the contents of Ethernet headers and their relationships to the encapsulated payload.

A.4 ARP

A.4.1 ARP Flood*

ARP Flood examines how the DUT handles unsolicited ARP replies. The test case generates large numbers of ARP replies and sends them to the DUT in an attempt to fill the ARP cache. The ARP cache is a memory table that maps IP and MAC addresses. Many devices accept unsolicited ARP replies and cache them regardless of their source.

A.4.2 ARP Grammar*

ARP Grammar describes and generates ARP request packets. Each packet contains valid and invalid header values including the source and destination IP addresses. Each request is mostly valid; this way, the DUT is more likely to process it. The grammar sends the packets over the Ethernet link to examine the DUT's ability to maintain both view and control while dealing with invalid packets.

A.5 IP

A.5.1 Random IP Packet Storm*

Random IP Packet Storm examines how the DUT handles large amounts of IP traffic that is not addressed to it. The test case simulates a situation in which the destination MAC address is broadcast or the device is listening in promiscuous mode. It generates large numbers of IP packets and sends them over the Ethernet link.

The destination IP address is randomly generated. Very few packets are addressed to the DUT. Subsequently, it ignores most of the test case traffic and the network stack processing cost is low.

The ability of a network interface card to handle IP packets varies with packet length, so it is useful to execute this test case several times using different packet length and rate limit values to determine the optimal values that the DUT can handle.

A.5.2 Unicast IP Packet Storm*

Unicast IP Packet Storm generates IP packets and sends them to the DUT at a specific packet rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

The test case generates identical packets. Each packet contains a null payload and all flags are set to zero. The source IP address is set to the satellite IP address and the destination IP address is set to the DUT IP address.

** Please contact Wurldtech for the complete test case description.*

A.5.3 Broadcast IP Packet Storm*

Broadcast IP Packet Storm generates IP packets and broadcasts them over the Ethernet link at a specific packet rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

The test case generates identical packets. Each packet contains a null payload and all flags are set to zero. The source IP address is set to the satellite IP address and the destination IP address is set to a broadcast IP address.

A.5.4 Multicast IP Packet Storm*

Multicast IP Packet Storm generates IP packets and multicasts them over the Ethernet link at a specific packet rate to examine the DUT's ability to maintain both view and control while dealing with the storm.

The test case generates identical packets. Each packet contains a null payload and all flags are set to zero. The source IP address is set to the satellite IP address and the destination IP address is set to a multicast IP address.

A.5.5 Fragmented IP Packet Storm*

Fragmented IP Packet Storm generates fragmented IP packets and sends them to the DUT at a specific packet rate to examine the DUT's ability to maintain both view and control while dealing with the storm. The test does not send the final fragment, in an attempt to force the DUT to keep incomplete packets in memory.

When an IP packet is larger than the maximum transmission unit (MTU) of a network segment over which it is to be sent, fragmentation is used to break the packet into smaller blocks of data. As the DUT must reassemble the original packet prior to processing and fragments may arrive out of order, it must keep the fragments in memory until the final fragment is received. However, the large number of incomplete fragmented packets sent during this test requires the DUT to discard incomplete fragmented packets to prevent memory and resource exhaustion.

The test case creates a packet that is the maximum IP packet size (65536 bytes) and fragments it to the maximum MTU size. The final fragment in the packet is not sent. The test case then increments the packet ID and creates the next packet. The protocol of each generated IP packet is set to UDP. The destination IP address is set to the DUT IP address.

A.5.6 IP Fuzzer*

IP Fuzzer generates IP packets using randomized header values and sends them to the DUT to examine the DUT's ability to maintain both view and control while dealing with the invalid and fragmented packets. The test case uses random values to generate invalid and fragmented packets.

This test case can be used to determine how the DUT handles different volumes of invalid traffic. For example, a DUT might handle a lot of packets containing few errors but fail if it receives a small number of packets containing many errors.

A.5.7 IP Grammar - Field Fuzzer*

IP Grammar - Field Fuzzer generates IP packets; each packet contains both valid and invalid header values. The grammar examines the DUT's ability to maintain both view and control while dealing with the invalid packets.

A.5.8 IP Grammar - Fragmentation*

IP Grammar - Fragmentation generates IP packets with invalid fragmentation and sends them to the DUT. The grammar examines the DUT's ability to maintain both view and control while processing or discarding invalidly fragmented packets.

** Please contact Wurldtech for the complete test case description.*

A.5.9 IP Grammar - Options Fields*

IP Grammar - Options Fields generates IP packets containing randomized option field values and sends them to the DUT. The grammar examines the DUT's ability to maintain both view and control while dealing with IP options.

The options field is rarely used. It is the final header field; it specifies security and routing information for the IP packet. Many IP stacks deployed in embedded devices completely ignore IP options or mistake IP options data as payload data.

A.6 ICMP

A.6.1 Destination Unreachable*

Destination Unreachable examines how the DUT handles ICMP Destination Unreachable messages. The grammar generates valid and invalid ICMP Destination Unreachable messages and sends them to the DUT.

A.6.2 Type/Code Cross Product*

Type/Code Cross Product generates ICMP packets with all type/code combinations that can be encoded in an ICMP header, including those that are not typically implemented or are not defined by RFC792, and sends each packet once to the DUT.

For each type/code combination, the test case sends packets of length 60, 1514 and the length specified in RFC792. There are known vulnerabilities in deployed ICMP implementations that are caused by poor range checking. This test case examines the DUT's ability to maintain both view and control while dealing with these invalid packets.

A.6.3 ICMP Grammar - Field Fuzzer*

ICMP Grammar - Field Fuzzer examines how the DUT handles message types that are not covered by specific ICMP grammar invocations. The grammar generates ICMP packets containing edge case values and non-standard message types and sends them to the DUT.

A.6.4 ICMP Grammar - Echo Request*

ICMP Grammar - Echo Request examines how the DUT handles edge case values for most fields in an ICMP echo request packet. The grammar generates ICMP echo request packets containing edge case values and sends them to the DUT.

An ICMP echo request packet checks network connectivity between two TCP/IP communication endpoints. It contains six fields; type, code, checksum, identifier, sequence number and implementation dependent data portion.

A.6.5 ICMP Grammar - Echo Response*

ICMP Grammar - Echo Response examines how the DUT handles edge case values for most fields in an ICMP echo response packet. The grammar generates ICMP echo response packets containing edge case values and sends them to the DUT.

n ICMP echo response message is sent to a system in response to an ICMP echo request message. Echo requests and replies are used to quickly check network connectivity between two endpoints. A response message contains six fields; type, code, checksum, identifier, sequence number and implementation dependent data portion.

** Please contact Wurldtech for the complete test case description.*

A.6.6 ICMP Grammar - Parameter Problem*

ICMP Grammar - Parameter Problem examines how the DUT handles edge case values for most fields in an ICMP parameter problem packet. The grammar generates ICMP parameter problem packets containing edge case values and sends them to the DUT.

A device sends an ICMP parameter problem packet to another device to tell it that there is a problem with the header fields in the packet it tried to send. The packet contains six fields; type, code, checksum, pointer, unused 16 bit field and the IP header. It also contains eight bytes of data from the undeliverable packet.

A.6.7 ICMP Grammar - Source Quench*

ICMP Grammar - Source Quench examines how the DUT handles edge case values for most fields in an ICMP source quench packet. The grammar generates ICMP source quench packets containing edge case values and sends them to the DUT.

A router under heavy load sends an ICMP source quench message to a device to tell it to slow down the rate of communication or use a different router to route the packet it is trying to send. The message contains five fields; type, code, checksum, an unused 32 bit field and the IP header. It also contains eight bytes of data from the misdirected packet.

A.6.8 ICMP Grammar - Redirect*

ICMP Grammar - Redirect examines how the DUT handles edge case values for most fields in an ICMP redirect packet. The grammar generates ICMP redirect packets containing edge case values and sends them to the DUT. A router sends an ICMP redirect packet to a device to tell it to redirect the packet it is trying to send to another system. The packet contains five fields; type, code, checksum, IP address and the IP header. It also contains eight bytes of data from the misdirected packet.

A.6.9 ICMP Grammar - Time Exceeded*

ICMP Grammar - Time Exceeded examines how the DUT handles edge case values for most fields in an ICMP time exceeded packet. The grammar generates ICMP time exceeded packets containing edge case values and sends them to the DUT.

A router sends an ICMP time exceeded packet to a device to tell it that the packet it was trying to send has timed out and is no longer routable or active. The packet contains five fields; type, code, checksum, an unused 32 bit field and the IP header. It also contains eight bytes of data from the misdirected packet.

A.6.10 ICMP Grammar - Timestamp*

ICMP Grammar - Timestamp examines how the DUT handles edge case values for most fields in ICMP timestamp request and reply messages. The grammar generates ICMP timestamp request packets containing edge case values and sends them to the DUT.

A device sends an ICMP timestamp request message to another device to ask it to send its current timestamp in an ICMP timestamp reply. In this way, two hosts can synchronise time. An ICMP timestamp request packet contains eight fields; type, code, checksum, identifier, sequence number and three timestamp fields. An ICMP timestamp reply message contains the same fields.

** Please contact Wurldtech for the complete test case description.*

A.7 TCP

A.7.1 TCP SYN Flood*

TCP SYN Flood generates SYN packets and sends them to the DUT at a specific packet rate to examine the DUT's ability to maintain both view and control while dealing with numerous new connection requests. As the test case does not send ACK packets, the new connections cannot be established.

The window size is 5000 and the SYN flag is set. The source IP address of each packet is set to an unused IP address and the destination IP address is set to the DUT IP address. Resource consumption can occur if the DUT does not discard incomplete packets.

A.7.2 TCP/IP LAND Attack*

TCP/IP LAND Attack generates TCP/IP packets; in each packet the source and destination address are the same and the source and destination port are the same. A LAND attack is a well known type of protocol attack which can cause a device to continuously reply to itself. This test case sends LAND packets to all open TCP ports and adjacent TCP ports on the DUT to examine the DUT's ability to maintain both view and control while dealing with the attack.

The test case generates identical packets. The SYN flag is set and the window size is 2048. The destination MAC address is set to the DUT MAC address. The source and destination IP addresses are set to the DUT IP address. The source and destination ports are equal.

A.7.3 TCP Fuzzer*

TCP Fuzzer generates TCP packets using randomized header values and sends them to the DUT to examine the DUT's ability to maintain both view and control while dealing with the fragmented and invalid TCP packets. The test case uses random values to generate invalid and fragmented packets.

This test case can be used to determine how the DUT handles different volumes of invalid traffic. For example, a DUT might handle a lot of packets containing few errors but fail if it receives a small number of packets containing many errors.

A.7.4 TCP Grammar - Field Fuzzer*

TCP Grammar - Field Fuzzer examines how the DUT handles edge case and bad field values. The grammar performs a basic fuzz of the 12 standard TCP header fields. These fields serve a variety of functions and can contain a variety of values. The grammar generates TCP packets containing edge case values for the header fields.

A.7.5 TCP Grammar - Options Fields*

TCP Grammar - Options Fields examines how the DUT handles TCP options. TCP packets can contain TCP options but many TCP stacks within embedded devices ignore them or assume that the length of the TCP header is constant.

A.7.6 TCP Grammar - Urgent Data*

TCP Grammar - Urgent Data examines how the DUT handles unauthorized packets containing an urgent flag. The urgent data flag indicates to TCP/IP stacks that the packet contains urgent data and should be given priority processing. The urgent data pointer points to the data in the TCP stream that corresponds to the urgent data. This mechanism works well only when a network administrator can control all the traffic and no unauthorized packets can have an urgent flag set.

** Please contact Wurldtech for the complete test case description.*

A.8 UDP

A.8.1 UDP Fuzzer*

UDP Fuzzer generates UDP packets using randomized header values and sends them to the DUT to examine the DUT's ability to maintain both view and control while dealing with the fragmented and invalid UDP packets. The test case uses random values to generate invalid and fragmented packets.

This test case can be used to determine how the DUT handles different volumes of invalid traffic. For example, a DUT might handle a lot of packets containing few errors but fail if it receives a small number of packets containing many errors.

A.8.2 UDP Grammar*

UDP Grammar generates UDP packets; each packet contains some valid and some invalid header values. The grammar examines the DUT's ability to maintain both view and control while dealing with the invalid packets.

** Please contact Wurldtech for the complete test case description.*

Wurldtech Security Technologies

For more information about Wurldtech Security Technologies contact:

Wurldtech Security Technologies, Inc.
401 W Georgia Street, Suite 1680
Vancouver BC Canada
V6B 5A1

Tel +1 (604) 669 6674
Fax +1 (604) 669 2902
Email info@wurldtech.com

We also invite you to visit our website at:
www.wurldtech.com