FORTRA™

DATASHEET (BEYOND SECURITY)

beSTORM

beSTORM is an intelligent black box fuzzer that ensures the security of products before they are released or deployed. It is built to meet the priorities of efficiency, flexibility, and breadth common to testing teams across the corporate landscape. beSTORM uses a proprietary prioritization algorithm to automatically start attacking the highest probability vulnerabilities first, before methodically expanding into billions of attacks across the chosen protocols. These protocols can be known, augmented, proprietary, or unknown.

beSTORM achieves this testing through almost no user interaction. It can be set up outside of the system being tested and can be scaled to use multiple processors or multiple machines to substantially reduce testing duration. It will automatically log anomalies and vulnerabilities and continues testing, eventually producing actionable reports that can be disseminated across teams and acted upon.

A Dynamic Testing Solution

Language Agnostic — beSTORM runs billions of attacks by testing the binary application and as such is completely indifferent of the programming language or system libraries used.

Scalable — beSTORM tests the entire communication range. This allows it to scale effortlessly with products with large complicated code bases.

Flexible — The lack of source code interaction also allows for exportation and use of beSTORM tests to teams that do not have access to the source code.

Uncovering the Unknown — Static code tools run thousands, or at best tens of thousands, of tests based on a certain set of case studies or scenarios of known vulnerabilities. This is equivalent to bolstering the product against known past attacks. beSTORM however, performs millions and potentially billions of attack combinations across the entire communication range. This tests the product against not only known past attacks but unknown future ones that may be launched against it.

"Beyond Security has built a lot of automation into its technology, but not at the expense of accuracy. At the end of the day, you're able to eradicate all vulnerabilities and stop things like backdoors and intruders from getting on the network."

–Brandon Buhai,
COO Beyond IP,
Chicago-based solution provider

Comprehensive

beSTORM is capable of testing both already known protocols and quickly learning and testing augmented, proprietary, or new protocols. It performs across all communication standards (even complex standards, such as SIP), and levels, including network, protocol, file, hardware, DLL, and API. beSTORM delivers an exhaustive search of all possible input combinations to test input implementation for weaknesses. During these tests it operates with a powerful monitor that detects and informs future attacks when even the slightest buffer overflow, format string or memory exception occurs even if these anomalies do not crash the system.

Flexible

beSTORM supports the analysis of proprietary or augmented protocols. If beSTORM encounters a proprietary or augmented protocol, beSTORM's Auto-Learn will log this and begin building and expanding tests into this protocol. These input types may be network traffic that beSTORM captured, a file sample that contains the network traffic captured using other means or a syntax describing an API. beSTORM uses these data sets to determine how the protocol or file is built. If the complete specification is available, users can also create or extend beSTORM modules using an XML editor.

beSTORM is also able to convert protocol standard text into an automated set of tests by converting the BNF description used in technical RFC documents into attack language. This allows beSTORM to be immediately updated and run on any new communication standards that are released by simply uploading a RFC document or the equivalent.

Actionable Reports

beSTORM is created with the end goal of remediation in mind. As such, beSTORM provides clean actionable reports which intricately detail encountered vulnerabilities that ended in a successful attack. Since beSTORM is behaviorally attacking the test product instead of testing against case studies, virtually all false positives are eliminated leaving the user with comprehensive and clean reports.

Found vulnerabilities can then be exported in a detailed vulnerability report that can be used to debug the application. Developers can load these vulnerability reports within their chosen development environment with zero knowledge of how to use beSTORM and immediately begin the debug process.

System Requirements

Recommended

- Quad-core processor (i5+ or equivalent)
- 8 GB of RAM (Windows 10)
- 1 GB available HDD space

beSTORM was designed to run on low power systems as well with the following

Bare minimum requirements

- x86-64 CPU
- 1 GB of RAM (Linux, Docker, Embedded application)
- 250 MB of HDD space

"We are very impressed with the beSTORM product. One notable feature is its flexibility in adding new and proprietary protocols. We are actively expanding the usage of beSTORM in our overall product portfolio as part of the standard security testing procedure."

–Avishai Avivi,
Sr. Director DPI Technologies Group,
Juniper Networks

Protocols

Basic

- DHCP Server (Simple)
- DNP3 - Master (Simple)
- DNP3 - Master Serial (Simple)
- DNP3 - Slave (Simple)
- DNP3 - Slave Serial (Simple)
- FTP Server (Simple)
- HTTP (Simple Web Client)
- HTTP Server (Simple Web Server)
- HTTPS Server (Simple Web Server)
- IEC61850 (MMS - Master - Simple)
- IEEE802.11 (AP - Simple)
- IEEE802.11 (Subscriber - Simple - UDP)
- IEEE802.11 (Subscriber - Simple)
- IMAP Server (Simple)
- Internet Printing Protocol (Simple)
- LLDP (Simple)
- Modbus (Master - Simple)
- Modbus (Master Serial - Simple)
- Session Initiation Protocol (Simple)
- SMPT Server (Simple)
- SNMP (Simple)
- SNMPv2 (Simple)
- SNMPv3 (Simple)
- SNMPv3 with MD5 (Simple)
- SNMPv3 with MD5 and DES (Simple)
- SNMPv3 with SHA1 (Simple)
- SSH Server (Simple)
- SSL Server (Simple)

Network / Client / TCP

- Border Gateway Protocol
- Common Industrial Protocol (EtherNet/IP)
- Diagnostic Over Internet Protocol
- Diameter
- DNP3 - Master (5 Layers)
- EtherNet/IP
- FTP
- HTTP/1.0 (SSL Web Client)
- HTTP/1.0 (Web Client)
- HTTP/1.1 (SSL Web Client)
- HTTP/1.1 (Web Client)
- HTTPS (Simple Web Client)
- ICAP
- IEC60870-5-104 - Master
- IEC61850 (MMS - MAster)
- IMAP
- KIES
- LDAP
- MCAP
- Modbus (Master)
- MQTTv3
- Network Configuration v1.0
- Network Configuration v1.1
- NFS Client
- NNTP
- OPC Unified Architecture - Simple
- Open Network Video Interface Forum
- Open vSwitch Database Management
- OpenFlow Switch
- POP3
- RMI (Client)
- RSH
- RTSP
- SFTP Client
- SMB Client
- SMTP
- SOAP over HTTP
- SSH
- Telnet
- TLS v1.2 Client
- TLS v1.3 Client
- TPKT

Network / Client / UDP

- IEEE802.11 (AP)
- IEEE802.11 (Subscriber)
- Advanced Audio Distribution Profile
- Alternate MAC/PHY
- AMQP
- ATT
- BFD Control
- BVLC
- COAP
- DHCP
- DNP3 - Master
- DNS

- DNS (Simple)
- Dropbox LAN Sync Discovery
- DVMRP
- EtherCAT over UDP
- Fastboot
- GTPv1
- HDCPv1.1
- HDCPv2.0
- HSU
- ISAKMP
- ISUP
- L2CAP
- Label Distribution Protocol (Simple)
- LLC
- LLMNR
- NAT-PMP
- NTP
- OBEX
- Portmap Client
- RADIUS
- Real-time Transport Protocol (with Register)
- Real-time Transport Protocol (without Register)
- RFCOMM
- RIPng
- RIPv1
- RTP Control Protocol
- Service Discovery Protocol
- Service Location Protocol
- Session Description Protocol
- Session Initiation Protocol
- Session Initiation Protocol (with Register)
- Simple Service Discovery Protocol
- STUN
- Syslog
- TAPA
- Teredo
- TFTP
- URB
- USB Mass Storage
- USB Request Block

Network / Server-side

- DNP3 - Slave
- DNS Server (Simple)
- IEC60870-5-104 - Slave
- IEC61850 (MMS - Slave)
- Modbus (Slave)
- NTP Server
- OpenFlow Controller

Low-level Network

- ISIS
- MPLS LDP
- OSPFv2
- OSPFv3
- AllJoyn
- ARP
- Cisco Discovery Protocol
- Cisco Group Management Protocol
- EtherCAT
- Ethernet
- Generic Routing Encapsulation
- GOOSE
- HSRP
- HSRPv2
- ICMPv4
- ICMPv6
- ICMPv6 (ND)
- IEEE802.1Q
- IGMPv0
- IGMPv1
- IGMPv2
- IGMPv3
- IPsec AH
- IPv4
- IPv6
- IPv6 (SHIM6)
- Layer 2 Tunneling Protocol
- Link Layer Discovery Protocol
- Locator/ID Separation Protocol
- PIMv2
- PPPoE

- PROFINET PTCP
- PROFINET RT
- PROFINET RT DCP
- RGMP
- SCTP
- STP (Spanning Tree Protocol)
- TCPv4
- TCPv6
- UDPlite
- UDPv4
- UDPv6
- VRRP

File

- ANI
- AVI (H264 -AC3)
- AVI (XVID)
- BMP
- DOC
- GIF
- HTML
- HWP
- ICO
- JASC-PAL
- JPEG
- MKV
- MP3
- MP4
- MPEG4
- PAL
- PCAP
- PDF
- PNG
- PPT
- TGA
- TIFF
- UPX
- WAV (PCM)
- WMV
- XLS

EDSA

- EDSAv2-401 Ethernet
- EDSAv2-402 ARP
- EDSAv2-403 IPv4
- EDSAv2-404 ICMPv4
- EDSAv2-405 UDPv4
- EDSAv2-406 TCPv4
- EDSAv3-401 Ethernet
- EDSAv3-402 ARP
- EDSAv3-403 IPv4
- EDSAv3-404 ICMPv4
- EDSAv3-405 UDPv4
- EDSAv3-406 TCPv4

Bluetooth / Bluetooth LE

- GAP
- GATT
- Hands free protocol
- HID over BT
- HID over GATT
- iBeacon
- Running Speed and Cadence

Serial

- DNP3 - Master Serial
- DNP3 - Slave Serial
- IEC60870-5-101 - Master
- IEC60870-5-101 - Slave
- Modbus (Master Serial)
- Modbus (Slave Serial)
- Modbus ASCII (Master Serial)
- Modbus RTU (Master Serial)
- Zigbee

Automotive

- CANBUS
- CANBUS (Over ICS)
- CANBUS (Over PCAN)
- CG4579 (Over PCAN)
- J1939-21
- OBDII

- OBDII (Over ICS)
- OBDII (Over PCAN)
- Unified Diagnostics Services
- UDS (Over ICS)
- UDS (Over PCAN)

Custom Development Module

For testing proprietary protocols, self-learning, and building your own protocols.

- TCPv4
- UDPv4

Fortra.com

About Fortra

Fortra is a cybersecurity company like no other. We're creating a simpler, stronger future for our customers. Our trusted experts and portfolio of integrated, scalable solutions bring balance and control to organizations around the world. We're the positive changemakers and your relentless ally to provide peace of mind through every step of your cybersecurity journey. Learn more at fortra.com.

