Trustwave®

Smart security on demand

Secure Web Gateway

MIB Data Breakdown

Legal Notice

While every precaution has been taken in the preparation of this document, Trustwave assumes no responsibility for errors or omissions. This publication and features described herein are subject to change without notice.

While the authors have used their best efforts in preparing this document, they make no representation or warranties with respect to the accuracy or completeness of the contents of this document and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the author nor Trustwave shall be liable for any loss of profit or any commercial damages, including but not limited to direct, indirect, special, incidental, consequential, or other damages.

The most current version of this document may be obtained by contacting:

Trustwave Technical Support:
Phone: +1.800.363.1621
Email: support@trustwave.com

Trademarks

Revision History

Version	Date	Changes
1.0	October 2013	First Trustwave version
2.0	November 2014	Update

Formatting Conventions

This manual uses the following formatting conventions to denote specific information.

Formats and Symbols	Meaning
Blue Underline	A blue underline indicates a Web site or email address.
Bold	Bold text denotes UI control and names such as commands, menu items, tab and field names, button and check box names, window and dialog box names, and areas of windows or dialog boxes.
`Code`	Text in `Courier New 9 pt in blue` indicates computer code or information at a command line.
Italics	Italics denotes the name of a published work, the current document, name of another document, text emphasis, to introduce a new term, and path names.
[Square brackets]	Square brackets indicate a placeholder for values and expressions.

Notes, Tips, and Cautions

🔍	**Note**: This symbol indicates information that applies to the task at hand.
💡	**Tip**: This symbol denotes a suggestion for a better or more productive way to use the product.
!	**Caution**: This symbol highlights a warning against using the software in an unintended manner.
?	**Question:** This symbol indicates a question that the reader should consider.

About This Guide

This document details the current Trustwave SWG MIB (Management Information Base) as at June 2014.

Table of Contents

1 General MIB information

The TrustwaveSWG node is 1.3.6.1.4.1.**6790.** TrustwaveSWG elements are under 1.3.6.1.4.1.6790.1.1 (trustwaveswg.products.vs).

A MIB definition file, called TRUSTWAVESWG-MIB.txt, is available from the Trustwave website. This file should be imported into your SNMP monitoring software.

2 SWG Device - General Information

General product nodes are under 1.3.6.1.4.1.6790.1.1 and are provided by the **Manager**:

- Product Version (2) – SWG product version
    - **Description**: The product version of the device
    - **Algorithm**: Locate "build_version" in the configuration file **etc** folder and extract the first three parts of the build number (For example, if the build number is "X.Y.Z.A", the product version will be "X.Y.Z")
- Build Information (3) – SWG Build Information
    - **Description**: The build number of the device
    - **Algorithm**: Locate "build_version" in the configuration file **etc** folder and set the value defined in it
- Device Type (10) – The Device type
    - **Description**: The SWG device role
    - **Algorithm**: Locate "role" in the configuration file **etc** folder and set the value defined in it
- Machine Type (11) – the Machine SWG type
    - **Description**: The SWG machine type
    - **Algorithm**: Locate "machinetype" in the configuration file **etc** folder and set the value defined in it

3 Scanning Information and Statistics

The following node values are provided by the **WASP**:

- Performance Stats (30) – Statistics regarding performance.
    - **Manager (5)** – Manager performance stats – **currently not in use!!**
    - **PS (6)** – Policy Server performance stats – **currently not in use!!**
    - **Log Handler (7)** – Log Client performance stats are detailed later on with more details. This part is **NOT** provided by the WASP.
    - **Scanner (20)** – This is a table with one leaf entry per WASP process. The leaf nodes are numbered consecutively.
        - BlockedReqs-per-second (1.1.2)
            - Type: 32 bit gauge
            - Description: Average rate of requests blocked per second
            - Algorithm: Calculate every 10 seconds the blocked requests per second in the last 60 seconds by dividing the number of requests that arrived in the last 60 seconds by 60. This is based on a sliding window algorithm in which the major frame is of 60 seconds and the minor one is of 10 seconds
        - Emergency Status (1.1.3) – Information regarding configured policies
            - Description: Current Emergency status
            - Algorithm: Checked the Emergency status as configured in FIDAL and set the value to ‘1’ if Emergency State is active or ‘0’ if it is not
        - Scanner Logged (2) – Table of Tables of logged messages. See WASP Nodes – Logged Node (20.2.1.1)
    - **ScannerProtocol (21)** – This is a table of tables. There is one table per protocol and within each item of the protocol, a table of wasp processes.
        - Last Reset Time (1.1.3) – Table of reset times
            - Description: The time in which all values provided by this protocol, per scanner were last reset.
        - Total Requests (1.1.4) – Table of total number of requests per protocol, per wasp process.
        - Average Total Requests per second (1.1.5) – Table of average total number of requests per protocol, per wasp process.
            - Average at 10 second intervals.

- Throughput-in-total (1.1.6) per protocol per wasp process
    - Type: 32 bit gauge
    - Description: Total input, in bytes, scanned since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if there is a result of the Direction CP and if the component size field has a valid value. If the Direction CP output is not “O”, the sub-node value is incremented by the component size
- Throughput-out-total (1.1.7) per protocol per wasp process
    - Type: 32 bit gauge
    - Description: Total output, in bytes, scanned since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if there is a result of the Direction CP and if the component size field has a valid value. If the Direction CP output is “O”, the sub-node value is incremented by the component size
- Blocked-total (1.1.8)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If a block reason is found this sub-node value is incremented by one
- Blocked-AV-total (1.19)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked due to Virus detection since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If so and its value is “Virus Detected”, this sub-node value is incremented by one

  - Blocked-BA-total (1.1.10)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked due to Behavior analysis since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If so and its value is “behavior ”, this sub-node value is incremented by one
  - Blocked-blacklist-total (1.1.11)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked due to being Blacklisted since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If so and its value is either “Access to this URL”, "The requested URL is an Adware site", "The requested URL is a Spyware site" or "Found item in a forbidden URL list", this sub-node value is incremented by one
  - Blocked-URLCat-total (1.1.12)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked due to URL category since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If so and its value is “Forbidden URL”, this sub-node value is incremented by one
  - Blocked-DLP-total (1.1.13)
    - Type: 64 bit counter
    - Description: Total number of requests which were blocked due to DLP since last reset (which is usually last time the WASP has started)
    - Algorithm: For each major item below the transaction level checks if the block reason is defined (block_reason). If so and its value is “data leakage”, this sub-node value is incremented by one
- Connection (40) – current total number of open HTTP and HTTPS connections. It has one sub-node – HTTP Connections (1) which of 32 bit Integer type. The Master Agent is configured to execute a script to get this sub-node value. This script is using one of the WASP backdoor functionalities to get the required answer.

4 WASP Protocols Table

- Independent of protocol (1) – Table by WASP process
  - Totals for all events
- HTTP (2) – Table by WASP process
  - Description: Totals for only HTTP events where the protocol is “HTTP”
- HTTPS (3) – Table by WASP process
  - Description: Totals for only HTTP events where the protocol is either “HTTPS”, “VS-SSL” or “HTTP Tunneling”
- FTP (4) – Table by WASP process
  - Description: Totals for only FTP events where the protocol is either “Native FTP” or “FTP over HTTP”
- ICAP (5) – Table by WASP process
  - Description: Totals for only ICAP events where the protocol is either “ICAP/HTTP”, “ICAP/HTTPS” or “ICAP/FTP over HTTP”

4.1 WASP Nodes – Logged Node (20.2.1.1)

All the data under the Logged node is calculated when the logging policy is being evaluated in the WASP business logic. The Logged node has the following sub-nodes:

- Logged-total (1)
  - Type: 64 bit counter
  - Description: Total number of requests which were logged since last reset (which is usually last time the WASP has started)
  - Algorithm: Always increment by one
- Logged-logsDb-total (2)
  - Type: 64 bit counter
  - Description: Total number of requests which were sent to the logging database since last reset (which is usually last time the WASP has started)
  - Algorithm: Increment by one if one of the logging action group ids is 202004
- Logged-archive-total (3)
  - Type: 64 bit counter
  - Description: Total number of requests which were sent to the archive system since last reset (which is usually last time the WASP has started)
  - Algorithm: Increment by one if one of the logging action group ids is 202007
- Logged-reportsDb-total (4)
  - Type: 64 bit counter
  - Description: Total number of requests which were sent to the reports database since last reset (which is usually last time the WASP has started)
  - Algorithm: Increment by one if one of the logging action group ids is 202008
- Logged-syslog-total (5)
  - Type: 64 bit counter
  - Description: Total number of requests which were sent to the syslog system since last reset (which is usually last time the WASP has started)
  - Algorithm: Increment by one if one of the logging action group ids is 202003

4.2 Log Client Information and Statistics

The Log Client provides the following sub-nodes under 1.3.6.1.4.1.6790.1.1.7:

- PID (1) – the Log Client process PID
- Last Reset Time (5) – The time in which all values provided by this process where last reset (usually the current process start time)
- Logged (12) – Actual stats for all logged transactions the passed through the Log Client, by their destination. This node has the following sub-nodes:
  - Logged-total (1)
    - Type: 64 bit counter
    - Description: Total number of log messages that were logged since last reset
    - Algorithm: Increment by one for each web log that is logged to any of the following four possible destinations
  - Logged-logsDb-total (2)
    - Type: 64 bit counter
    - Description: Total number of log messages that were inserted to the web logs database since last reset
    - Algorithm: Increment by one if inserted a message to the logs DB
  - Logged-archive-total (3)
    - Type: 64 bit counter
    - Description: Total number of log messages that were written to an archive file since last reset
    - Algorithm: Increment by one for each web log that is written to an archive file
  - Logged-reportsDb-total (4)
    - Type: 64 bit counter
    - Description: Total number of log messages that were inserted to the reports database since last reset
    - Algorithm: Increment by one if inserted a message to the reports DB
  - Logged-syslog-total (5)
    - Type: 64 bit counter
    - Description: Total number of log messages that were sent to a syslog server since last reset

Algorithm: Increment by one for each web log that is sent to all configures syslog servers

About Trustwave

Trustwave is a leading provider of compliance, Web, application, network and data security solutions delivered through the cloud, managed security services, software and appliances. For organizations faced with today's challenging data security and compliance environment, Trustwave provides a unique approach with comprehensive solutions that include its TrustKeeper® portal and other proprietary security solutions. Trustwave has helped hundreds of thousands of organizations—ranging from Fortune 500 businesses and large financial institutions to small and medium-sized retailers—manage compliance and secure their network infrastructures, data communications and critical information assets. Trustwave is headquartered in Chicago with offices worldwide. For more information, visit https://www.trustwave.com.