



FortiAnalyzer and FortiGate

Version 4.0 MR2
SQL Log Database Query Technical Note

FortiAnalyzer™ and FortiGate™ SQL Log Database Query Technical Note

Version 4.0 MR2

10 June 2010

Revision 5

© Copyright 2010 Fortinet, Inc. All rights reserved. No part of this publication including text, examples, diagrams or illustrations may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical or otherwise, for any purpose, without prior written permission of Fortinet, Inc.

Trademarks

Dynamic Threat Prevention System (DTPS), APSecure, FortiASIC, FortiBIOS, FortiBridge, FortiClient, FortiGate®, FortiGate Unified Threat Management System, FortiGuard®, FortiGuard-Antispam, FortiGuard-Antivirus, FortiGuard-Intrusion, FortiGuard-Web, FortiLog, FortiAnalyzer, FortiManager, Fortinet®, FortiOS, FortiPartner, FortiProtect, FortiReporter, FortiResponse, FortiShield, FortiVoIP, and FortiWiFi are trademarks of Fortinet, Inc. in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Regulatory compliance

FCC Class A Part 15 CSA/CUS



Caution: Risk of explosion if battery is replaced by incorrect type.
Dispose of used batteries according to instructions.

Contents

Introduction	5
Registering your Fortinet product.....	5
Customer service & technical support	5
Training	5
Documentation	6
Conventions	6
IP addresses.....	6
Cautions, Notes and Tips	6
Typographical conventions.....	7
Command syntax conventions.....	7
Querying FortiAnalyzer SQL log databases	11
Creating datasets	11
Troubleshooting	14
SQL tables	14
Log severity levels	17
Log fields in each table	17
Common log fields.....	17
Application control log fields	19
Attack log fields.....	21
DLP archive / content log fields	22
Data Leak Prevention log fields.....	27
Email filter log fields.....	28
Event log fields	29
Malform Description Values	39
Traffic log fields.....	43
Antivirus log fields.....	45
Web filter log fields	47
Netscan log fields	48
Examples	49
Example 1: Distribution of applications by type in the last 24 hours.....	51
GUI procedure.....	51
CLI procedure	51
Notes:.....	51
Example 2: Top 100 applications by bandwidth in the last 24 hours	52
GUI procedure.....	52
CLI procedure	52
Notes:.....	52
Example 3: Top 10 attacks in the past one hour	53

GUI procedure.....	53
CLI procedure	53
Notes:.....	53
Example 4: Top WAN optimization applications in the past 24 hours	53
GUI procedure.....	53
CLI procedure	54
Querying FortiGate SQL log databases	55
Creating datasets	55
SQL tables	56
Log severity levels	57
Examples	58
Example 1: Distribution of Applications by Type in the last 24 hours	59
CLI commands	59
Notes:.....	59
Example 2: Top 10 Application Bandwidth Usage Per Hour Summary	60
CLI commands	60
Notes:.....	60
Example 3: Top 10 Attacks Over The Last 24 Hours	60
CLI commands	60
Notes:.....	60
Example 4: Wan Optimization Application in LAN Composition over Last 24 Hours	61
CLI commands	61
Notes:.....	61

Introduction

Welcome and thank you for selecting Fortinet products for your network protection.

FortiAnalyzer units support local PostgreSQL and remote MySQL databases for storage of log tables. FortiGate units with hard disks support local SQLite databases for storage of log tables.

This document describes how to write your own SQL query statements to create custom datasets and describes the fields in each type of log table to assist in writing SQL queries.

This document supplements both the FortiAnalyzer Administration Guide and the FortiGate Administration Guide.

This section contains the following topics:

- [Registering your Fortinet product](#)
- [Customer service & technical support](#)
- [Training](#)
- [Documentation](#)
- [Conventions](#)

Registering your Fortinet product

Before you begin configuring and customizing features, take a moment to register your Fortinet product at the Fortinet Technical Support web site, <https://support.fortinet.com>.

Many Fortinet customer services, such as firmware updates, technical support, and FortiGuard Antivirus and other FortiGuard services, require product registration.

For more information, see the Fortinet Knowledge Base article [Registration Frequently Asked Questions](#).

Customer service & technical support

Fortinet Technical Support provides services designed to make sure that you can install your Fortinet products quickly, configure them easily, and operate them reliably in your network.

To learn about the technical support services that Fortinet provides, visit the Fortinet Technical Support web site at <https://support.fortinet.com>.

You can dramatically improve the time that it takes to resolve your technical support ticket by providing your configuration file, a network diagram, and other specific information. For a list of required information, see the Fortinet Knowledge Base article [Fortinet Technical Support Requirements](#).

Training

Fortinet Training Services provides classes that orient you quickly to your new equipment, and certifications to verify your knowledge level. Fortinet provides a variety of training programs to serve the needs of our customers and partners world-wide.

To learn about the training services that Fortinet provides, visit the Fortinet Training Services web site at <http://campus.training.fortinet.com>, or email them at training@fortinet.com.

Documentation

The Fortinet Technical Documentation web site, <http://docs.fortinet.com>, provides the most up-to-date versions of Fortinet publications, as well as additional technical documentation such as technical notes.

In addition to the Fortinet Technical Documentation web site, you can find Fortinet technical documentation on the Fortinet Tools and Documentation CD, and on the Fortinet Knowledge Base.

Fortinet Tools and Documentation CD

Many Fortinet publications are available on the Fortinet Tools and Documentation CD shipped with your Fortinet product. The documents on this CD are current at shipping time. For current versions of Fortinet documentation, visit the Fortinet Technical Documentation web site, <http://docs.fortinet.com>.

Fortinet Knowledge Base

The Fortinet Knowledge Base provides additional Fortinet technical documentation, such as troubleshooting and how-to-articles, examples, FAQs, technical notes, and more. Visit the Fortinet Knowledge Base at <http://kb.fortinet.com>.

Comments on Fortinet technical documentation

Please send information about any errors or omissions in this technical document to techdoc@fortinet.com.

Conventions

Fortinet technical documentation uses the conventions described below.

IP addresses

To avoid publication of public IP addresses that belong to Fortinet or any other organization, the IP addresses used in Fortinet technical documentation are fictional and follow the documentation guidelines specific to Fortinet. The addresses used are from the private IP address ranges defined in RFC 1918: Address Allocation for Private Internets, available at <http://ietf.org/rfc/rfc1918.txt?number-1918>.

Cautions, Notes and Tips

Fortinet technical documentation uses the following guidance and styles for cautions, notes and tips.



Caution: Warns you about commands or procedures that could have unexpected or undesirable results including loss of data or damage to equipment.



Note: Presents useful information, usually focused on an alternative, optional method, such as a shortcut, to perform a step.



Tip: Highlights useful additional information, often tailored to your workplace activity.

Typographical conventions

Fortinet documentation uses the following typographical conventions:

Table 1: Typographical conventions in Fortinet technical documentation

Convention	Example
Button, menu, text box, field, or check box label	From <i>Minimum log level</i> , select <i>Notification</i> .
CLI input	<pre>config system dns set primary <address_ipv4> end</pre>
CLI output	<pre>FGT-602803030703 # get system settings comments : (null) opmode : nat</pre>
Emphasis	HTTP connections are <i>not</i> secure and can be intercepted by a third party.
File content	<pre><HTML><HEAD><TITLE>Firewall Authentication</TITLE></HEAD> <BODY><H4>You must authenticate to use this service.</H4></pre>
Hyperlink	Visit the Fortinet Technical Support web site, https://support.fortinet.com .
Keyboard entry	Type a name for the remote VPN peer or client, such as <code>Central_Office_1</code> .
Navigation	Go to <code>VPN > IPSEC > Auto Key (IKE)</code> .
Publication	For details, see the FortiGate Administration Guide .

Command syntax conventions

The command line interface (CLI) requires that you use valid syntax, and conform to expected input constraints. It will reject invalid commands.

Brackets, braces, and pipes are used to denote valid permutations of the syntax. Constraint notations, such as `<address_ipv4>`, indicate which data types or string patterns are acceptable value input.

Table 2: Command syntax notation

Convention	Description
Square brackets []	A non-required word or series of words. For example: <code>[verbose {1 2 3}]</code> indicates that you may either omit or type both the <code>verbose</code> word and its accompanying option, such as: <code>verbose 3</code>

Table 2: Command syntax notation

<p>Angle brackets < ></p>	<p>A word constrained by data type.</p> <p>To define acceptable input, the angled brackets contain a descriptive name followed by an underscore (_) and suffix that indicates the valid data type. For example:</p> <p><retries_int></p> <p>indicates that you should enter a number of retries, such as 5.</p> <p>Data types include:</p> <ul style="list-style-type: none"> • <xxx_name>: A name referring to another part of the configuration, such as policy_A. • <xxx_index>: An index number referring to another part of the configuration, such as 0 for the first static route. • <xxx_pattern>: A regular expression or word with wild cards that matches possible variations, such as *@example.com to match all email addresses ending in @example.com. • <xxx_fqdn>: A fully qualified domain name (FQDN), such as mail.example.com. • <xxx_email>: An email address, such as admin@mail.example.com. • <xxx_url>: A uniform resource locator (URL) and its associated protocol and host name prefix, which together form a uniform resource identifier (URI), such as http://www.fortinet.com/. • <xxx_ipv4>: An IPv4 address, such as 192.168.1.99. • <xxx_v4mask>: A dotted decimal IPv4 netmask, such as 255.255.255.0. • <xxx_ipv4mask>: A dotted decimal IPv4 address and netmask separated by a space, such as 192.168.1.99 255.255.255.0. • <xxx_ipv4/mask>: A dotted decimal IPv4 address and CIDR-notation netmask separated by a slash, such as 192.168.1.99/24. • <xxx_ipv6>: A colon (:)-delimited hexadecimal IPv6 address, such as 3f2e:6a8b:78a3:0d82:1725:6a2f:0370:6234. • <xxx_v6mask>: An IPv6 netmask, such as /96. • <xxx_ipv6mask>: An IPv6 address and netmask separated by a space. • <xxx_str>: A string of characters that is not another data type, such as P@ssw0rd. Strings containing spaces or special characters must be surrounded in quotes or use escape sequences. See the FortiWeb CLI Reference. • <xxx_int>: An integer number that is not another data type, such as 15 for the number of minutes.
<p>Curly braces { }</p>	<p>A word or series of words that is constrained to a set of options delimited by either vertical bars or spaces.</p> <p>You must enter at least one of the options, unless the set of options is surrounded by square brackets [].</p>

Table 2: Command syntax notation

	Options delimited by vertical bars 	Mutually exclusive options. For example: {enable disable} indicates that you must enter either <code>enable</code> or <code>disable</code> , but must not enter both.
	Options delimited by spaces	Non-mutually exclusive options. For example: {http https ping snmp ssh telnet} indicates that you may enter all or a subset of those options, in any order, in a space-delimited list, such as: ping https ssh Note: To change the options, you must re-type the entire list. For example, to add <code>snmp</code> to the previous example, you would type: ping https snmp ssh If the option adds to or subtracts from the existing list of options, instead of replacing it, or if the list is comma-delimited, the exception will be noted.

Querying FortiAnalyzer SQL log databases

The FortiAnalyzer unit supports local PostgreSQL and remote MySQL databases for storage of log tables.

To create a report based on the FortiGate log messages in a local or remote database, you can use either the predefined datasets, or create your own custom datasets by querying the log messages in the SQL database.

This document describes the procedure for creating datasets, and describes the fields in each type of log table to assist in writing SQL queries.

This section contains the following topics:

- [Creating datasets](#)
- [SQL tables](#)
- [Examples](#)

Creating datasets

The following procedure describes how to create datasets in the web-based manager. You can also use the CLI command `config sql-report dataset` to create datasets. For details, see the [FortiAnalyzer CLI Reference](#) and the “[Examples](#)” section.

To create a custom data set in the web-based manager

- 1 Go to *Report > Chart > Data Set*.
- 2 Click *Create New*.
- 3 Configure the following, then click *OK*.

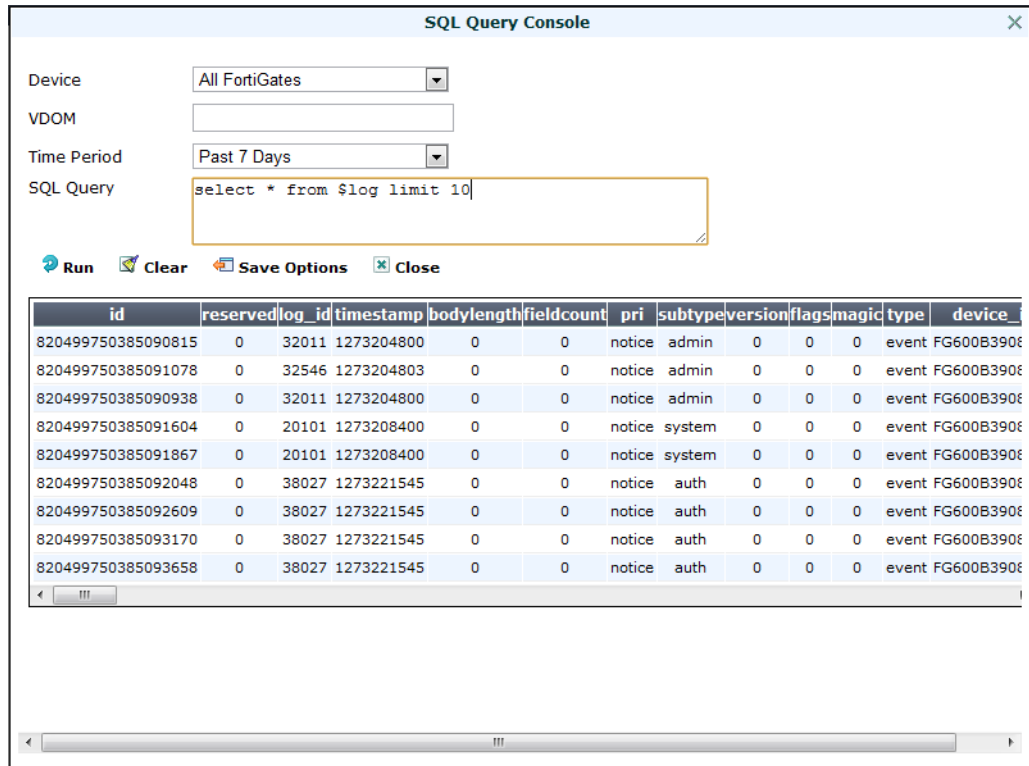
Name of the GUI item	Description
Name	Enter the name for the data set.
Log Type (\$log)	Enter the type of logs to be used for the data set. <i>\$log</i> is used in the SQL query to represent the log type you select, and it is run against all tables of this type.

Time Period (\$filter)	Select to use logs from a time frame, or select <i>Specified</i> and define a custom time frame by selecting the <i>Begin Time</i> and <i>End Time</i> . <i>\$filter</i> is used in the SQL query "where" clause to limit the results to the period you select.
Past N Hours/Days/Weeks	If you selected <i>Past N Hours/Days/Weeks</i> for <i>Time Period</i> , enter the number.
Begin Time	Enter the date (or use the calendar icon) and time of the beginning of the custom time range. This option appears only when you select <i>Specified</i> in the Time Period (\$time) field.
End Time	Enter the date (or use the calendar icon) and time of the end of the custom time range. This option appears only when you select <i>Specified</i> in the Time Period (\$time) field.
SQL Query	Enter the SQL query syntax to retrieve the log data you want from the SQL database. Different SQL systems use different query syntaxes to deal with date/time format. The FortiAnalyzer unit uses PostgreSQL as the local database and supports MySQL as the remote database. To facilitate querying in both MySQL and PostgreSQL systems, you can use the following default date/time macros and query syntaxes for the corresponding time period you choose: <ul style="list-style-type: none"> • <i>Hour_of_day</i>: For example, you can select <i>Yesterday</i> for the <i>Time Period</i> and enter the syntax "select \$hour_of_day as hourstamp, count(*) from \$log where \$filter group by hourstamp order by hourstamp". • <i>Day_of_week</i>: For example, you can select <i>This Week</i> for the <i>Time Period</i> and enter the syntax "select \$day_of_week as datestamp, count(*) from \$log where \$filter group by datestamp order by datestamp". • <i>Day_of_month</i>: For example, you can select <i>This Month</i> for the <i>Time Period</i> and enter the syntax "select \$day_of_month as datestamp, count(*) from \$log where \$filter group by datestamp order by datestamp". • <i>Week_of_year</i>: For example, you can select <i>This Year</i> for the <i>Time Period</i> and enter the syntax "select \$week_of_year as weekstamp, count(*) from \$log where \$filter group by weekstamp order by weekstamp". • <i>Month_of_year</i>: For example, you can select <i>This Year</i> for the <i>Time Period</i> and enter the syntax "select \$month_of_year as monthstamp, count(*) from \$log where \$filter group by monthstamp order by monthstamp". <p>The results of running the queries will display the date and time first, followed by the log data.</p>
Test	Click to test whether or not the SQL query is successful. See "To test a SQL query" on page 12 .

To test a SQL query

- 1 Follow the procedures in ["To create a custom data set in the web-based manager" on page 11](#).
- 2 After entering the SQL query, click *Test*.
- 3 Configure the following, then click *Run* to view the query results.

Figure 1: SQL Query test results



Name of the GUI Description item

Device	Select a specific FortiGate unit, FortiMail unit, or FortiClient installation, or select all devices, to apply the SQL query to.
VDom	If you want to apply the SQL query to a FortiGate VDOM, enter the name of the VDOM.
Time Period (\$filter)	Select to query the logs from a time frame, or select <i>Specified</i> and define a custom time frame by selecting the <i>Begin Time</i> and <i>End Time</i> . <i>\$filter</i> is used in the where clause of the SQL query to limit the results to the period you select.
Past N Hours/Days/Weeks	If you selected <i>Past N Hours/Days/Weeks</i> for <i>Time Period</i> , enter the number.
Begin Time	Enter the date (or use the calendar icon) and time of the beginning of the custom time range. This option appears only when you select <i>Specified</i> in the Time Period (<i>\$filter</i>) field.
End Time	Enter the date (or use the calendar icon) and time of the end of the custom time range. This option appears only when you select <i>Specified</i> in the Time Period (<i>\$filter</i>) field.
SQL Query	Enter the SQL query to retrieve the log data you want from the SQL database.
Run	Click to execute the SQL query. The results display. If the query is not successful, see “Troubleshooting” on page 14 .
Clear	Select to remove the displayed query results.

Save Options	Select to save the SQL query console configuration to the data set configuration. The Device and VDOM configurations are not used by the data set configuration.
Close	Click to return to the data set configuration page.

Troubleshooting

If the query is unsuccessful, an error message appears in the results window indicating the cause of the problem.

SQL statement syntax errors

Here are some example error messages and possible causes:

You have an error in your SQL syntax (remote/MySQL) or ERROR: syntax error at or near... (local/PostgreSQL)

- Check that SQL keywords are spelled correctly, and that the query is well-formed.
- Table and column names are demarked by grave accent (`) characters. Single (') and double (") quotation marks will cause an error.

No data is covered.

- The query is correctly formed, but no data has been logged for the log type. Check that you have configured the FortiAnalyzer unit to save that log type. Under *System > Config > SQL Database*, make sure that the log type is checked.

Connection problems

If well formed queries do not produce results, and logging is turned on for the log type, there may be a database configuration problem with the remote database.

Ensure that:

- MySQL is running and using the default port 3306.
- You have created an empty database and a user with create permissions for the database.

Here is an example of creating a new MySQL database named fazlogs, and adding a user for the database:

```
#Mysql -u root -p
mysql> Create database fazlogs;
mysql> Grant all privileges on fazlogs.* to 'fazlogger'@'*'
identified by 'fazpassword';
mysql> Grant all privileges on fazlogs.* to
'fazlogger'@'localhost' identified by 'fazpassword';
```

SQL tables

The FortiAnalyzer™ and FortiGate™ unit creates a database table for each managed device and each log type, when there is log data. If the FortiAnalyzer unit is not receiving data from a device, or logging is not enabled under *System > Config > SQL Database*, it does not create log tables for that device.

SQL tables follow the naming convention of [Device Name]-[SQL table type]-[timestamp], where the SQL table type is one of the types listed in [Table 3 on page 15](#).



Note: The timestamp portion of the log name depends on the FortiAnalyzer unit firmware release. It is either the creation time of the table (in releases before 4.2.1), or the timestamp of the log on disk (in releases 4.2.1 and later).

To view all the named tables created in a database, you can use:

- local (PostgreSQL) database: `SELECT * FROM pg_tables`
- remote (MySQL): `SHOW TABLES`

The names of all created tables and their types are stored in a master table named `table_ref`.

Table 3: Log types and table types

Log Type	SQL table type	Description
Traffic log	tlog	The traffic log records all traffic to and through the FortiGate interface.
Event log	elog	The event log records management and activity events. For example, when an administrator logs in or logs out of the web-based manager.
Antivirus log	vlog	The antivirus log records virus incidents in Web, FTP, and email traffic.
Webfilter log	wlog	The web filter log records HTTP FortiGate log rating errors including web content blocking actions that the FortiGate unit performs.
Attack log	alog	The attack log records attacks that are detected and prevented by the FortiGate unit.
Spamfilter log	slog	The spam filter log records blocking of email address patterns and content in SMTP, IMAP, and POP3 traffic.
Data Leak Prevention log	dlog	The Data Leak Prevention log records log data that is considered sensitive and that should not be made public. This log also records data that a company does not want entering their network.
Application Control log	rlog	The application control log records data detected by the FortiGate unit and the action taken against the network traffic depending on the application that is generating the traffic, for example, instant messaging software, such as MSN Messenger.
DLP archive log	clog	The DLP archive log, or <code>clog.log</code> , records all log messages, including most IM log messages as well as the following session control protocols (VoIP protocols) log messages: <ul style="list-style-type: none"> • SIP start and end call • SCCP phone registration • SCCP call info (end of call) • SIMPLE log message
Vulnerability Management log	nlog	The vulnerability management log, or <code>netscan.log</code> , contains logging events generated by a network scan.

FortiAnalyzer™ and FortiGate™ logs also include log sub-types, which are types of log messages that are within the main log type. For example, in the event log type there are the subtype admin log messages. FortiAnalyzer™ and FortiGate™ log types and subtypes are numbered, and these numbers appear within the log identification field of the log message.

Table 4: Log Sub-types

Log Type	Sub-Type
traffic (Traffic Log)	<ul style="list-style-type: none"> allowed – Policy allowed traffic violation – Policy violation traffic Other
event (Event Log)	<p>For FortiGate devices:</p> <ul style="list-style-type: none"> system – System activity event ipsec – IPSec negotiation event dhcp – DHCP service event ppp – L2TP/PPTP/PPPoE service event admin – admin event ha – HA activity event auth – Firewall authentication event pattern – Pattern update event alertemail – Alert email notifications chassis – FortiGate-4000 and FortiGate-5000 series chassis event sslvpn-user – SSL VPN user event sslvpn-admin – SSL VPN administration event sslvpn-session – SSL VPN session even his-performance – performance statistics vipssl – VIP SSL events ldb-monitor – LDB monitor events
dlp (Data Leak Prevention)	<ul style="list-style-type: none"> dlp – Data Leak Prevention
app-crtl (Application Control Log)	<ul style="list-style-type: none"> app-crtl-all – All application control
DLP archive (DLP Archive Log)	<ul style="list-style-type: none"> HTTP – Virus infected FTP – FTP content metadata SMTP – SMTP content metadata POP3 – POP3 content metadata IMAP – IMAP content metadata
virus (Antivirus Log)	<ul style="list-style-type: none"> infected – Virus infected filename – Filename blocked oversize – File oversized
webfilter (Web Filter Log)	<ul style="list-style-type: none"> content – content block urlfilter – URL filter FortiGuard block FortiGuard allowed FortiGuard error ActiveX script filter Cookie script filter Applet script filter
ips (Attack Log)	<ul style="list-style-type: none"> signature – Attack signature anomaly – Attack anomaly
emailfilter (Spam Filter Log)	<ul style="list-style-type: none"> SMTP POP3 IMAP

Log severity levels

You can define what severity level the FortiGate unit records logs at when configuring the logging location. The FortiGate unit logs all message at and above the logging severity level you select. For example, if you select Error, the unit logs Error, Critical, Alert, and Emergency level messages.

Table 5: Log Severity Levels

Levels	Description	Generated by
0 - Emergency	The system has become unstable.	Event logs, specifically administrative events, can generate an emergency severity level.
1 - Alert	Immediate action is required.	Attack logs are the only logs that generate an Alert severity level.
2 - Critical	Functionality is affected.	Event, Antivirus, and Spam filter logs.
3 - Error	An error condition exists and functionality could be affected.	Event and Spam filter logs.
4 - Warning	Functionality could be affected.	Event and Antivirus logs.
5 - Notification	Information about normal events.	Traffic and Web Filter logs.
6 - Information	General information about system operations.	Content Archive, Event, and Spam filter logs.

The Debug severity level, not shown in [Table 5](#), is rarely used. It is the lowest log severity level and usually contains some firmware status information that is useful when the FortiGate unit is not functioning properly. Debug log messages are only generated if the log severity level is set to Debug. Debug log messages are generated by all types of FortiGate features.

Log fields in each table

This section describes the fields of each log table stored in an SQL database. Because of differences in SQL dialects, some fields have different types depending on whether they are stored locally or remotely.

The tables described in this section are:

- [“Common log fields,” on page 17](#)
- [“Application control log fields” on page 19](#)
- [“Attack log fields” on page 21](#)
- [“DLP archive / content log fields” on page 22](#)
- [“Data Leak Prevention log fields” on page 27](#)
- [“Email filter log fields” on page 28](#)
- [“Event log fields” on page 29](#)
- [“Traffic log fields” on page 43](#)
- [“Antivirus log fields” on page 45](#)
- [“Web filter log fields” on page 47](#)
- [“Netscan log fields” on page 48](#)

Common log fields

All log tables share some common fields, described in [Table 6](#).

Table 6: Common Fields

Field	Type		Description	Tables
	PostgreSQL	MySQL		
id	int not null primary key	int unsigned not null primary key	ID / primary key for the record	all
itime	timestamp	datetime	The time the log event was received by the FortiAnalyzer.	all
dtime	timestamp	datetime	The time the log event was generated on the device.	all
cluster_id	varchar(24)	varchar(24)	The HA cluster ID if the FortiGate runs in HA mode.	all
device_id	varchar(16)	varchar(16)	The serial number of the device.	all
log_id	int default 0	smallint unsigned default 0	A ten-digit number. The first two digits represent the log type and the following two digits represent the log subtype. The last one to five digits are the message id. For more detail about what the combination of type, subtype and message ID means, see the FortiGate Log Message Reference.	all
subtype	varchar(255)	varchar(255)	The subtype of the log message. The possible values of this field depend on the log type. See Table 4 for a list of subtypes associated with each log type.	all
type	varchar(255)	varchar(255)	The log type.	all
timestamp	int default 0	int unsigned default 0	Timestamp for the event	all
pri	varchar(255)	varchar(255)	The log priority level. See Table 5 for a list of priority levels and the log types that generate them.	all
vd	varchar(255)	varchar(255)	The virtual domain where the traffic was logged. If no virtual domains are enabled and configured, this field contains the virtual domain, root.	all
user	varchar(255)	varchar(255)	The name of the user creating the traffic.	all except nlog
group	varchar(255)	varchar(255)	The name of the group creating the traffic.	all except nlog
src	varchar(40) (255 for alog)	varchar(40) (255 for alog)	The source IP address.	all except nlog
dst	varchar(40) (255 for alog)	varchar(40) (255 for alog)	The destination IP address.	all except nlog
src_port	int default 0	smallint unsigned default 0	The source port of the TCP or UDP traffic. The source protocol is zero for other types of traffic.	all except nlog
dst_port	int default 0	smallint unsigned default 0	The destination port number of the TCP or UDP traffic. The destination port is zero for other types of traffic.	all except nlog
src_int	varchar(255)	varchar(255)	The interface where the through traffic comes in. For outgoing traffic originating from the firewall, it is "unknown".	all except clog and nlog
dst_int	varchar(255)	varchar(255)	The interface where the through traffic goes to the public or Internet. For incoming traffic to the firewall, it is "unknown".	all except clog and nlog
policyid	bigint default 0	int unsigned default 0	The ID number of the firewall policy that applies to the session or packet. Any policy that is automatically added by the FortiGate will have an index number of zero. For more information, see the Fortinet Knowledge Base article, Firewall policy=0.	all except nlog

Table 6: Common Fields

service	varchar(255)	varchar(255)	The service of where the activity or event occurred, whether it was on a web page using HTTP or HTTPs. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • http • https • smtp • pop3 • imap • ftp • mm1 • mm3 • mm4 • mm7 • nntp • im • smtps • pop3s • imaps 	all except clog
identidx	bigint default 0	int unsigned default 0	The identity index number.	all except nlog
profile	varchar(255)	varchar(255)	The protection profile associated with the firewall policy that traffic used when the log message was recorded.	all except dlog, tlog, and nlog
profiletype	varchar(255)	varchar(255)	The type of profile associated with the firewall policy that traffic used when the log message was recorded.	all except dlog, tlog, and nlog
profilegroup	varchar(255)	varchar(255)	The profile group associated with the firewall policy that traffic used when the log message was recorded.	all except dlog, tlog, and nlog

Application control log fields

The table below lists the fields defined in application control log tables (type rlog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For application control logs, this field can be: <ul style="list-style-type: none"> • request • cancel • accept • fail • download • stop • start • end • timeout • blocked • succeeded • failed • authentication-required • pass • block
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display MSISDN of the phone that sent the MMS message. This field will always display N/A in FortiOS.

Field	Type		Description
	PostgreSQL	MySQL	
kind	varchar(255)	varchar(255)	This field is an enum, and can be one of the following values: <ul style="list-style-type: none"> login chat file photo audio call regist unregister call-block request response
dir	varchar(255)	varchar(255)	The direction of the traffic. This field is an enum, and can be one of the following: <ul style="list-style-type: none"> incoming outgoing N/A
src_name	varchar(255)	varchar(255)	The name of the source or the source IP address.
dst_name	varchar(255)	varchar(255)	The destination name or destination IP address.
proto	int default 0	smallint unsigned default 0	The protocol number that applies to the session or packet. The protocol number in the packet header that identifies the next level protocol. Protocol number's are assigned by the Internet Assigned Number Authority (IANA).
serial	bigint default 0	int unsigned default 0	Serial number of the log message.
app_list	varchar(255)	varchar(255)	The application control list (under <i>UTM > Application Control > Application Control List</i> on the FortiGate unit) that contains the policy that triggered this log item.
app_type	varchar(255)	varchar(255)	The application category.
app	varchar(255)	varchar(255)	The application name. You can look the application type up in <i>UTM > Application Control > Application List</i> , and then select the name that is in the field to go to more detailed information on the FortiGuard Encyclopedia.
action	varchar(255)	varchar(255)	The action the FortiGate unit took for this session or packet. This field is an enum and can be one of the following values: <ul style="list-style-type: none"> pass block monitor kickout encrypt-kickout reject
count	bigint default 0	int unsigned default 0	Total number of blocked applications.
filename	varchar(255)	varchar(255)	The file name associated with the blocked application.
filesize	bigint default 0	int unsigned default 0	The file size of the file.
message	varchar(255)	varchar(255)	The blocked message of chat applications.
content	varchar(255)	varchar(255)	Content of the blocked applications.

Field	Type		Description
	PostgreSQL	MySQL	
reason	varchar(255)	varchar(255)	The reason why the log was recorded. This field is an enum, and can be one of the following values: <ul style="list-style-type: none"> meter-overload-drop meter-overload-refuse rate-limit dialog-limit long-header unrecognized-form unknown block-request invalid-ip exceed-rate
req	varchar(255)	varchar(255)	Request.
phone	varchar(255)	varchar(255)	Phone number of the blocked application.
msg	varchar(255)	varchar(255)	Explains why the log was recorded.
attack_id	bigint default 0	int unsigned default 0	Attack ID.

Attack log fields

The table below lists the fields defined in attack log tables (type alog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For attack logs, this field can be: <ul style="list-style-type: none"> detected dropped reset reset_client reset_server drop_session pass_session clear_session
serial	bigint default 0	int unsigned default 0	The serial number of the log message.
attack_id	bigint default 0	int unsigned default 0	The identification number of the attack log message.
severity	varchar(255)	varchar(255)	The specified severity level of the attack. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> info low medium high critical
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always display N/A.
sensor	varchar(255)	varchar(255)	The DLP sensor that was used.

Field	Type		Description
	PostgreSQL	MySQL	
icmp_id	varchar(255)	varchar(255)	The Internet Control Message Protocol (ICMP) message ID (returned for ECHO REPLY).
icmp_type	varchar(255)	varchar(255)	The ICMP message type.
icmp_code	varchar(255)	varchar(255)	The ICMP message code.
proto	smallint default 0	tinyint unsigned default 0	The protocol of the event.
ref	varchar(255)	varchar(255)	A reference URL to the Fortiguard IPS database for more information about the attack.
count	bigint default 0	int unsigned default 0	The number of times that attack was detected within a short period of time. This is useful when the attacks are DoS attacks.
incident_serialno	bigint default 0	int unsigned default 0	The unique ID for this attack. This number is used for cross-references IPS packet logs.
msg	varchar(255)	varchar(255)	Explains the activity or event that the FortiGate unit recorded. In this example, an attack occurred that could have caused a system crash.

DLP archive / content log fields

The table below lists the fields defined in application DLP / Content log tables (type clog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred.
clogver	smallint default 0	tinyint unsigned default 0	The version of the content log.
epoch	bigint default 0	int unsigned default 0	The unique number for each archive. It is used for cross reference purposes.
eventid	bigint default 0	int unsigned default 0	The ID of the archive event.
SN	bigint default 0	int unsigned default 0	The session number.
endpoint	varchar(255)	varchar(255)	The ID of the endpoint, such as MSISDN or account ID.
client	varchar(40)	varchar(40)	The IP of the client.
server	varchar(40)	varchar(40)	The IP of the server.
laddr	varchar(40)	varchar(40)	The local IP.
raddr	varchar(40)	varchar(40)	The remote IP.

Field	Type		Description
	PostgreSQL	MySQL	
cstatus	varchar(255)	varchar(255)	The cstatus field can be any one of the following: <ul style="list-style-type: none"> • clean • infected • heuristic • banned_word • blocked • exempt • oversize • carrier_endpoint_filter (FortiOS Carrier only) • mass_mms (FortiOS Carrier only) • dlp • fragmented • spam • im_summary • im-message • im_file_request (a file was transferred) • im_file_accept (an file was accepted) • im_file_cancel • im_voice (an IM voice chat) • im_photo_share_request (a photo was shared) • im_photo_share_cancel • im_photo_share_stop • im_photo_xfer (a photo was transferred during the chat) • voip • error

Field	Type		Description
	PostgreSQL	MySQL	
infection	varchar(255)	varchar(255)	The infection type. This field is an enum, and can be one of the following: <ul style="list-style-type: none"> • bblock • fileexempt • file intercept • mms block • carrier end point filter • mms flood • mms duplicate • virus • virusrm • heuristic • html script • script filter • banned word • exempt word • oversize • virus • heuristic • worm • mime block • fragmented • exempt • ip blacklist • dnsbl • FortiGuard - AntiSpam ip blacklist • helo • emailblacklist • mimeheader • dns • FortiGuard - AntiSpam ase block • banned word • ipwhitelist • emailwhitelist • fewwhitelist • headerwhitelist • wordwhitelist • dlp • dlpban • pass • mms content checksum
virus	varchar(255)	varchar(255)	The virus name.
rcvd	bigint default 0	int unsigned default 0	The number of bytes that were received from the client.
sent	bigint default 0	int unsigned default 0	The number of bytes that were received from the server.
method	varchar(255)	varchar(255)	The type of HTTP command used. For example, GET.
url	varchar(255)	varchar(255)	The URL address of the web site that was accessed.
cat	varchar(255)	varchar(255)	The http/https category.
cat_desc	varchar(255)	varchar(255)	The http/https category description.
to	varchar(255)	varchar(255)	To
from	varchar(255)	varchar(255)	From
subject	varchar(255)	varchar(255)	Subject
direction	varchar(255)	varchar(255)	Incoming or outgoing.

Field	Type		Description
	PostgreSQL	MySQL	
attachment	smallint default 0	tinyint unsigned default 0	Mail attachment present.
ftpcmd	varchar(255)	varchar(255)	The FTP command. This field is an enum and can be one of: <ul style="list-style-type: none"> • NONE • USER • PASS • ACCT • STOR • RETR • QUIT
file	varchar(255)	varchar(255)	The archive file name.
local	varchar(255)	varchar(255)	The local user.
remote	varchar(255)	varchar(255)	The remote user.
proto	varchar(255)	varchar(255)	The protocol.
kind	varchar(255)	varchar(255)	The kind field can be any one of the following: <ul style="list-style-type: none"> • summary • chat • file (a file was transferred) • photo (photo sharing) • photo-xref (a photo was transferred) • audio (a voice chat) • oversize (an oversized file) • fileblock (a file was blocked) • fileexempt • virus • dlp • call-block (SIP call blocked) • call-info (SIP call information) • call (SIP call) • register (SIP register) • unregister (SIP unregister)
action	varchar(255)	varchar(255)	The action.
dir	varchar(255)	varchar(255)	The direction, either "inbound" or "outbound".
messages	bigint default 0	int unsigned default 0	The message number.
start-date	varchar(255)	varchar(255)	The local start date.
end-date	varchar(255)	varchar(255)	The local end date.
content	varchar(255)	varchar(255)	IM chat content.
filename	varchar(255)	varchar(255)	File name.
filesize	bigint default 0	int unsigned default 0	File size.
message	varchar(255)	varchar(255)	Message.
conn-mode	varchar(255)	varchar(255)	Connection mode.
heuristic	varchar(255)	varchar(255)	Heuristic.
duration	bigint default 0	int unsigned default 0	The duration of the session.
reason	varchar(255)	varchar(255)	The reason.
phone	varchar(255)	varchar(255)	Phone number.
dlp_sensor	varchar(255)	varchar(255)	DLP sensor.

Field	Type		Description
	PostgreSQL	MySQL	
message_type	varchar(255)	varchar(255)	The message type. This field is an enum, and be one of: <ul style="list-style-type: none"> request response
request_name	varchar(255)	varchar(255)	Request name.
malform_desc	varchar(255)	varchar(255)	Malformed content description. This field is an enum, and can be one of the values listed in Table 7 on page 26 .
malform_data	bigint default 0	int unsigned default 0	Malform data.
line	varchar(255)	varchar(255)	Line.
column	bigint default 0	int unsigned default 0	Column.

Table 7: Values for malform-desc

<att-field>-expected	<att-value>-expected	<bandwidth>-expected	<bwtype>-execpted	<callid>-expected	<CSeq-num>-expected
<delta-seconds>-expected	<encoding-name>-expected-in-rtpmap	<fmt>-expected	<gen-value>-expected	<generic-param>-with-invalid-<gen-value>	<integer>-expected
<m-attribute>-expected-after-SEMI	<m-subtype>-expected	<m-type>-expected	<media>-expected	<method>-does-not-match-the-request-line	<method>-expected
<Method>-expected-after-<CSeq-num>	<payload-type>-expected-in-rtpmap	<proto>-expected	<repeat-interval>-expected	<response-num>-expected	<seq>-number-expected
<sess-id>-expected	<sess-version>-expected	<text>-expected	<time>-expected	<token>-expected-in-<proto>-after-slash	<typed-time>-expected
<username>-exepcted	<word>-expected	boundary-parameter-appears-more-than-once	colon-expected	digits-expected	domain-label-oversize
domain-name-invalid	domain-name-oversize	duplicated-sip-header	empty-quoted-string	end-of-line-error	EQUAL-expected-after-<m-attribute>
expires-header-repeated	header-line-oversize	header-parameter-expected	IN-expected	invalid-<clock-rate>-in-rtpmap	invalid-<encoding-parameters>-in-rtpmap
invalid-<gen-value>	invalid-<m-value>	invalid-<protocol-name>	invalid-<protocol-version>	invalid-<quoted-string>-in-<gen-value>	invalid-<quoted-string>-in-<m-value>
invalid-<SIP-Version>-on-request-line	invalid-<start-time>	invalid-<stop-time>	invalid-<transport>	invalid-<userinfo>	invalid-branch-parameter
invalid-candidate-line	invalid-escape-encoding-in-<reason-phrase>	invalid-escape-encoding-in-<userinfo>	invalid-escape-encoding-in-uri-header	invalid-escape-encoding-in-uri-parameter	invalid-expires-parameter
invalid-fqdn	invalid-ipv4-address	invalid-ipv6-address	invalid-maddr-parameter	invalid-max-forwards	invalid-method-uri-parameter
invalid-port	invalid-port-after-ip-address-in-alt-line	invalid-port-after-ip-address-in-candidate-line	invalid-port-in-rtcp-line	invalid-q-parameter	invalid-quoted-string-in-display-name
invalid-quoting-character	invalid-received-parameter	invalid-rport-parameter	invalid-status-code	invalid-tag-parameter	invalid-transport-uri-parameter
invalid-ttl-parameter	invalid-ttl-uri-parameter	invalid-uri-header-name	invalid-uri-header-name-value-pair	invalid-uri-header-value	invalid-uri-parameter-pname

Table 7: Values for malform-desc

invalid-uri-parameter-value	invalid-user-uri-parameter	IP-expected	IP4-or-IP6-expected	ipv4-address-expected	IPv4-or-IPv6-address-expected
ipv6-address-expected	left-angle-bracket-is-mandatory	line-order-error	LWS-expected	missing-mandatory-field	msg-body-oversize
multipart-Content-Type-has-no-boundary	no-matching-double-quote	no-METHOD-on-request-line	no-SLASH-after-<protocol-name>	no-SLASH-after-<protocol-version>	no-tag-parameter
o-line-not-allowed-on-media-level	port-expected	port-not-allowed	r-line-not-allowed-on-media-level	right-angle-bracket-not-found	s-line-not-allowed-on-media-level
sdp-alt-line-before-m-line	sdp-candidate-line-before-m-line	sdp-invalid-alt-line	sdp-rtcp-line-before-m-line	sdp-v-o-s-t-lines-are-mandatory	sip-udp-message-truncated
sip-Yahoo-candidate-invalid-protocol	slash-expected-after-<encoding-name>-in-rtmpmap	SLASH-expected-after-<m-type>	space-violation	syntax-malformed	t-line-not-allowed-on-media-level
token-expected	too-many-c-lines	too-many-candidate-lines	too-many-i-lines	too-many-m-lines	too-many-o-lines
too-many-rtcp-lines	too-many-s-lines	too-many-v-line	trailing-bytes	unexpected-character	unknown-header
unknown-scheme	uri-expected	uri-parameter-repeat	uri-parameters-not-allowed-by-RFC	v-line-not-allowed-on-media-level	via-parameter-repeat
whitespace-expected	z-line-not-allowed-on-media-level				

Data Leak Prevention log fields

The table below lists the fields defined in data leak prevention log tables (type dlog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For DLP logs, this field can be: <ul style="list-style-type: none"> • detected • blocked
service	varchar(255)	varchar(255)	The service of where the activity or event occurred. For DLP logs, this field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • http • https • smtp • pop3 • imap • ftp • mm1 • mm3 • mm4 • mm7 • nntp • im • smtps • pop3s • imaps
serial	bigint default 0	int unsigned default 0	The serial number of the log message.

Field	Type		Description
	PostgreSQL	MySQL	
sport	int default 0	smallint unsigned default 0	The source port.
dport	int default 0	smallint unsigned default 0	The destination port.
hostname	varchar(255)	varchar(255)	The host name or IP address.
url	varchar(255)	varchar(255)	The URL address of the web site that was visited.
from	varchar(255)	varchar(255)	The sender's email address.
to	varchar(255)	varchar(255)	The receiver's email address.
msg	varchar(255)	varchar(255)	Explains the activity or event that the FortiGate unit recorded.
rulename	varchar(255)	varchar(255)	The name of the rule within the DLP sensor.
compoundname	varchar(255)	varchar(255)	The compound name.
action	varchar(255)	varchar(255)	The action that was specified within the rule. In some rules within sensors, you can specify content archiving. If no log type is specified, this field displays log-only. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> log-only block exempt ban ban sender quarantine ip quarantine interface
severity	smallint default 0	tinyint unsigned default 0	The level of severity for the specified rule.

Email filter log fields

The table below lists the fields defined in email filter log tables (type slog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For email filter logs, this field can be: <ul style="list-style-type: none"> exempted blocked detected

Field	Type		Description
	PostgreSQL	MySQL	
service	varchar(255)	varchar(255)	The service of where the activity or event occurred. For DLP logs, this field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • http • smtp • pop3 • imap • ftp • mm1 • mm3 • mm4 • mm7 • im • nntp • https • smtps • imaps • pop3s
serial	bigint default 0	int unsigned default 0	The serial number of the log message.
sport	int default 0	smallint unsigned default 0	The source port.
dport	int default 0	smallint unsigned default 0	The destination port.
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always displays N/A.
from	varchar(255)	varchar(255)	The sender's email address.
to	varchar(255)	varchar(255)	The receiver's email address.
banword	varchar(255)	varchar(255)	The name of the Banned Word policy.
tracker	varchar(255)	varchar(255)	Tracker
dir	varchar(255)	varchar(255)	The email direction. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • tx • rx
agent	varchar(255)	varchar(255)	This field is for FortiGate units running FortiOS Carrier. If you do not have FortiOS Carrier running on your FortiGate unit, this field always displays N/A.
msg	varchar(255)	varchar(255)	Explains the activity or event that the FortiGate unit recorded. In this example, the sender's email address is in the blacklist and matches the fourth email address in that list.

Event log fields

The table below lists the fields defined in event log tables (type elog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	<p>The status of the action the FortiGate unit took when the event occurred.</p> <p>For event logs, the possible values of this field depend on the subcategory:</p> <p>subcategory ipsec</p> <ul style="list-style-type: none"> • success • failure • negotiate_error • esp_error • dpd_failure <p>subcategory voip</p> <ul style="list-style-type: none"> • start • end • timeout • blocked • succeeded • failed • authentication-required <p>subcategory gtp</p> <ul style="list-style-type: none"> • forwarded • prohibited • rate-limited • state-invalid • tunnel-limited • traffic-count • user-data
msg	varchar(255)	varchar(255)	Explains the activity or event that the FortiGate unit recorded.
ssid	varchar(255)	varchar(255)	The service set identifier.

Field	Type		Description
	PostgreSQL	MySQL	
action	varchar(255)	varchar(255)	<p>The action the FortiGate unit should take for this firewall policy.</p> <p>For event logs, the possible values of this field depend on the subcategory of the event:</p> <p>subcategory ipsec:</p> <ul style="list-style-type: none"> • negotiate • error • install_sa • delete_phase1_sa • delete_ipsec_sa • dpd • tunnel-up • tunnel-down • tunnel-stats • phase2-up • phase2-down <p>subcategory nac-quarantine:</p> <ul style="list-style-type: none"> • ban-ip • ban-interface • ban-src-dst-ip <p>subcategory sslvpn-user</p> <ul style="list-style-type: none"> • tunnel-up • tunnel-down • ssl-login-fail <p>subcategory sslvpn-admin</p> <ul style="list-style-type: none"> • info <p>subcategory sslvpn-session</p> <ul style="list-style-type: none"> • tunnel-stats • ssl-web-deny • ssl-web-pass • ssl-web-timeout • ssl-web-close • ssl-sys-busy • ssl-cert • ssl-new-con • ssl-alert • ssl-exit-fail • ssl-exit-error • tunnel-up • tunnel-down • tunnel-statsssl-tunnel-unknown-tag • ssl-tunnel-error

Field	Type		Description
	PostgreSQL	MySQL	
action (continued)			<p>subcategory voip:</p> <ul style="list-style-type: none"> • permit • block • monitor • kickout • encrypt-kickout • cm-reject • exempt • ban • ban-user • log-only <p>subcategory his-performance</p> <ul style="list-style-type: none"> • perf-stats
session_id	bigint default 0	int unsigned default 0	The session ID
count	bigint default 0	int unsigned default 0	The number of dropped SIP packets.
proto	varchar(255)	varchar(255)	The protocol
cpu	smallint default 0	tinyint unsigned default 0	The CPU usage, for performance.
epoch	bigint default 0	int unsigned default 0	The unique number for each archive. It is used for cross reference purposes.
mem	smallint default 0	tinyint unsigned default 0	The memory usage, for performance.
duration	bigint default 0	int unsigned default 0	The duration of the interval for item counts (such as infected, scanned, etc) in this log entry.
infected	bigint default 0	int unsigned default 0	The number of infected messages.
from	varchar(255)	varchar(255)	Source IP address.
ha_group	smallint default 0	tinyint unsigned default 0	High availability group
tunnel_id	bigint default 0	int unsigned default 0	Tunnel ID
bssid	varchar(255)	varchar(255)	The basic service set identifier.
tunnel_type	varchar(255)	varchar(255)	Tunnel type
event_id	bigint default 0	int unsigned default 0	Event ID
ip	varchar(40)	varchar(40)	IP address
ha_role	varchar(255)	varchar(255)	High availability role.
rem_ip	varchar(40)	varchar(40)	Remote IP (used in ipsec subcategory logs).
suspicious	bigint default 0	int unsigned default 0	The number of suspicious messages.
sn	varchar(255)	varchar(255)	Serial number of the event
to	varchar(255)	varchar(255)	Destination IP address..
total_session	bigint default 0	int unsigned default 0	Total IP sessions.
ap	varchar(255)	varchar(255)	The physical AP name.
scanned	bigint default 0	int unsigned default 0	The number of scanned messages.
vcluster	bigint default 0	int unsigned default 0	Virtual cluster.
remote_ip	varchar(40)	varchar(40)	Remote IP (Used in sslvpn-* subcategory logs).
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always displays N/A.

Field	Type		Description
	PostgreSQL	MySQL	
imsi	varchar(255)	varchar(255)	An International Mobile Subscriber Identity or IMSI is a unique number associated with all GSM and UMTS network mobile phone users.
loc_ip	varchar(40)	varchar(40)	Local IP
from_vcluster	bigint default 0	int unsigned default 0	From virtual cluster.
rem_port	int default 0	smallint unsigned default 0	Remote port.
msisdn	varchar(255)	varchar(255)	The MSISDN of the carrier endpoint.
tunnel_ip	varchar(40)	varchar(40)	Tunnel IP.
intercepted	bigint default 0	int unsigned default 0	The number of intercepted messages.
vap	varchar(255)	varchar(255)	The virtual AP name.
apn	varchar(255)	varchar(255)	The access point name.
out_intf	varchar(255)	varchar(255)	The out interface.
blocked	bigint default 0	int unsigned default 0	The number of blocked messages.
mac	varchar(255)	varchar(255)	MAC address.
to_vcluster	bigint default 0	int unsigned default 0	To virtual cluster.
acct_stat	varchar(255)	varchar(255)	The accounting state. This is an enum and can have one of the following values: <ul style="list-style-type: none"> • Start • Stop • Interim-Update • Accounting-On • Accounting-Off
selection	varchar(255)	varchar(255)	The selection. This is an enum and can have one of the following values: <ul style="list-style-type: none"> • apns-vrf • ms-apn-no-vrf • net-apn-no-vrf
reason	varchar(255)	varchar(255)	The reason this log was generated.
rate	smallint default 0	tinyint unsigned default 0	Traffic rate
loc_port	int default 0	smallint unsigned default 0	Local port.
vcluster_member	bigint default 0	int unsigned default 0	Virtual cluster member.
vcluster_state	varchar(255)	varchar(255)	Virtual cluster state.
app-type	varchar(255)	varchar(255)	Application type.
nsapi	smallint default 0	tinyint unsigned default 0	Network Service Access Point Identifier, an identifier used in cellular data networks.
dport	int default 0	smallint unsigned default 0	Destination port.
channel	smallint default 0	tinyint unsigned default 0	Channel.
cookies	varchar(255)	varchar(255)	Cookies.
checksum	bigint default 0	int unsigned default 0	The number of content checksum blocked messages.
dst_host	varchar(255)	varchar(255)	Destination host name or IP.

Field	Type		Description
	PostgreSQL	MySQL	
nf_type	varchar(255)	varchar(255)	The notification type. This is an enum and can have one of the following values: <ul style="list-style-type: none"> • bword • file_block • carrier_ep_bwl • flood • dupe • alert • mms_checksum • virus
vdname	varchar(255)	varchar(255)	The VDOM name.
linked-nsapi	smallint default 0	tinyint unsigned default 0	Linked Network Service Access Point Identifier.
next_stats	bigint default 0	int unsigned default 0	Next Statistics.
virus	varchar(255)	varchar(255)	Virus name.
imei-sv	varchar(255)	varchar(255)	International Mobile Equipment Identity or IMEI is a number, usually unique, to identify GSM, WCDMA, and iDEN mobile phones, as well as some satellite phones.
devintfname	varchar(255)	varchar(255)	The device interface name.
security	varchar(255)	varchar(255)	The wireless security. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • open • wep64 • wep128 • wpa-psk • wpa-radius • wpa • wpa2 • wpa2-auto
policy_id	bigint default 0	int unsigned default 0	The policy ID that triggered this log.
rai	varchar(255)	varchar(255)	Routing Area Identification.
hostname	varchar(255)	varchar(255)	The host name or IP
xauth_user	varchar(255)	varchar(255)	Authenticated user name.
uli	varchar(255)	varchar(255)	User Location Information.
xauth_group	varchar(255)	varchar(255)	Authenticated user group.
sent	numeric(20) default 0	bigint unsigned default 0	Number of bytes sent.
rcvd	numeric(20) default 0	bigint unsigned default 0	Number of bytes received.
sess_duration	bigint default 0	int unsigned default 0	The duration of the session.
hbdn_reason	varchar(255)	varchar(255)	Heartbeat down reason. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • linkfail • neighbor-info-lost
banned_src	varchar(255)	varchar(255)	Banned source. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • ips • dos • dlp-rule • dlp-compound • av

Field	Type		Description
	PostgreSQL	MySQL	
end-usr-address	varchar(40)	varchar(40)	End user address.
msg-type	smallint default 0	tinyint unsigned default 0	Message type.
sync_type	varchar(255)	varchar(255)	Synchronization type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • configurations • external-files
banned_rule	varchar(255)	varchar(255)	Banned rule / reason.
vpn_tunnel	varchar(255)	varchar(255)	VPN tunnel.
sync_status	varchar(255)	varchar(255)	Synchronization status. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • out-of-sync • in-sync
alert	varchar(255)	varchar(255)	Alert.
sensor	varchar(255)	varchar(255)	Sensor name.
endpoint	varchar(255)	varchar(255)	The endpoint.
stage	smallint default 0	tinyint unsigned default 0	Stage.
voip_proto	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • sip • sccp
deny_cause	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • packet-sanity • invalid-reserved-field • reserved-msg • out-state-msg • reserved-ie • out-state-ie • invalid-msg-length • invalid-ie-length • miss-mandatory-ie • ip-policy • non-ip-policy • sgsn-not-authorized • sgsn-no-handover • ggsn-not-authorized • invalid-seq-num • msg-filter • apn-filter • imsi-filter • adv-policy-filter
desc	varchar(255)	varchar(255)	Description
dir	varchar(255)	varchar(255)	Direction (inbound or outbound).
kind	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • register • unregister • call • call-info • call-block

Field	Type		Description
	PostgreSQL	MySQL	
init	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • local • remote
mode	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • aggressive • main • quick • xauth • xauth_client
cert-type	varchar(255)	varchar(255)	Certificate type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • CA • CRL • Local • Remote
ui	varchar(255)	varchar(255)	User interface.
exch	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • NSA_INIT • AUTH • CREATE_CHILD
rat-type	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • utran • geran • wlan • gan • hspa
error_num	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • Invalid ESP packet detected. • Invalid ESP packet detected (HMAC validation failed). • Invalid ESP packet detected (invalid padding). • Invalid ESP packet detected (invalid padding length). • Invalid ESP packet detected (replayed packet). • Received ESP packet with unknown SPI.
method	varchar(255)	varchar(255)	The method.
phase2_name	varchar(255)	varchar(255)	IPSec VPN Phase 2 name
spi	varchar(255)	varchar(255)	IPSec VPN SPI.
c-sgsn	varchar(40)	varchar(40)	SGSN IP address for GTP signalling.
request_name	varchar(255)	varchar(255)	Request name
seq	varchar(255)	varchar(255)	Sequence number
c-ggsn	varchar(40)	varchar(40)	GGSN IP address for GTP signalling.
in_spi	varchar(255)	varchar(255)	Remote SPI in IPSec VPN configuration.
u-sgsn	varchar(40)	varchar(40)	SGSN IP address for GTP user traffic.
out_spi	varchar(255)	varchar(255)	Local SPI in IPSec VPN configuration.
u-ggsn	varchar(40)	varchar(40)	GGSN IP address for GTP user traffic.

Field	Type		Description
	PostgreSQL	MySQL	
c-sgsn-teid	bigint default 0	int unsigned default 0	SGSN TEID (Tunnel endpoint identifier) for signalling.
enc_spi	varchar(255)	varchar(255)	Encryption SPI in IPsec VPN.
c-ggsn-teid	bigint default 0	int unsigned default 0	GGSN TEID for signalling.
dec_spi	varchar(255)	varchar(255)	Decryption SPI in IPsec VPN.
message_type	varchar(255)	varchar(255)	Message type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> request response
malform_desc	varchar(255)	varchar(255)	Malformed description. This field is an enum. See "Malform Description Values" on page 39 for possible values.
tunnel	varchar(255)	varchar(255)	Tunnel name
u-sgsn-teid	bigint default 0	int unsigned default 0	SGSN TEID for user traffic.
u-ggsn-teid	bigint default 0	int unsigned default 0	GGSN TEID for user traffic.
malform_data	bigint default 0	int unsigned default 0	Malformed data.
tunnel-idx	bigint default 0	int unsigned default 0	VPN tunnel index.
line	varchar(255)	varchar(255)	The content of malformed SIP line.
column	bigint default 0	int unsigned default 0	The syntax error point in the SIP line.
c-pkts	numeric(20) default 0	bigint unsigned default 0	Number of packets for signalling.
phone	varchar(255)	varchar(255)	SCCP phone device name.
profile_group	varchar(255)	varchar(255)	Profile group name.
c-bytes	numeric(20) default 0	bigint unsigned default 0	Number of bytes for signalling.
u-pkts	numeric(20) default 0	bigint unsigned default 0	Number of packets used for traffic.
profile_type	varchar(255)	varchar(255)	Profile type.
u-bytes	numeric(20) default 0	bigint unsigned default 0	Number of bytes used for traffic.
next_stat	bigint default 0	int unsigned default 0	Next stat.
user_data	varchar(255)	varchar(255)	User data.
role	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> responder initiator
result	varchar(255)	varchar(255)	This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> ERROR OK DONE PENDING
xauth_result	varchar(255)	varchar(255)	Authorization result. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> XAUTH authentication successful XAUTH authentication failed

Field	Type		Description
	PostgreSQL	MySQL	
esp_transform	varchar(255)	varchar(255)	ESP Transform. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • ESP_NULL • ESP_DES • ESP_3DES • ESP_AES
esp_auth	varchar(255)	varchar(255)	ESP Authorization. This field is an enum, and can have one of the following values: no authentication <ul style="list-style-type: none"> • HMAC_SHA1 • HMAC_MD5 • HMAC_SHA256
error_reason	varchar(255)	varchar(255)	Text explanation for the error. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • invalid certificate • invalid SA payload • probable preshared key mismatch • peer SA proposal not match local policy • peer notification • not enough key material for tunnel • encapsulation mode mismatch • no matching gateway for new request • aggressive vs main mode mismatch for new request

Field	Type		Description
	PostgreSQL	MySQL	
peer_notif	varchar(255)	varchar(255)	<p>Peer Notification. This field is an enum, and can have one of the following values:</p> <ul style="list-style-type: none"> • NOT-APPLICABLE • INVALID-PAYLOAD-TYPE • DOI-NOT-SUPPORTED • SITUATION-NOT-SUPPORTED • INVALID-COOKIE • INVALID-MAJOR-VERSION • INVALID-MINOR-VERSION • INVALID-EXCHANGE-TYPE • INVALID-FLAGS • INVALID-MESSAGE-ID • INVALID-PROTOCOL-ID • INVALID-SPI • INVALID-TURN-ID • ATTRIBUTES-NOT-SUPPORTED • NO-PROPOSAL-CHOSEN • BAD-PROPOSAL-SYNTAX • PAYLOAD-MALFORMED • INVALID-KEY-INFORMATION • INVALID-ID-INFORMATION • INVALID-CERT-ENCODING • INVALID-CERTIFICATE • BAD-CERT-REQUEST-SYNTAX • INVALID-CERT-AUTHORITY • INVALID-HASH-INFORMATION • AUTHENTICATION-FAILED • INVALID-SIGNATURE • ADDRESS-NOTIFICATION • NOTIFY-SA-LIFETIME • CERTIFICATE-UNAVAILABLE • UNSUPPORTED-EXCHANGE-TYPE • UNEQUAL-PAYLOAD-LENGTHS • CONNECTED • RESPONDER-LIFETIME • REPLAY-STATUS • INITIAL-CONTACT • R-U-THERE • R-U-THERE-ACK • HEARTBEAT • RETRY-LIMIT-REACHED

Malform Description Values

- unexpected-character
- invalid-quoting-character
- trailing-bytes
- header-line-oversize
- msg-body-oversize
- domain-name-oversize
- domain-label-oversize
- syntax-malformed

- duplicated-sip-header
- space-violation
- invalid-ipv4-address
- invalid-ipv6-address
- invalid-port
- invalid-fqdn
- no-matching-double-quote
- empty-quoted-string
- invalid-<userinfo>
- invalid-escape-encoding-in-<userinfo>
- invalid-escape-encoding-in-uri-parameter
- invalid-escape-encoding-in-uri-header
- invalid-escape-encoding-in-<reason-phrase>
- port-expected
- port-not-allowed
- domain-name-invalid
- <gen-value>-expected
- invalid-<gen-value>
- invalid-<quoted-string>-in-<gen-value>
- ipv4-address-expected
- ipv6-address-expected
- uri-expected
- invalid-transport-uri-parameter
- invalid-user-uri-parameter
- invalid-method-uri-parameter
- invalid-ttl-uri-parameter
- invalid-uri-parameter-pname
- invalid-uri-parameter-value
- uri-parameter-repeat
- invalid-uri-header-name
- invalid-uri-header-value
- invalid-uri-header-name-value-pair
- invalid-quoted-string-in-display-name
- left-angle-bracket-is-mandatory
- right-angle-bracket-not-found
- invalid-status-code
- no-METHOD-on-request-line
- uri-parameters-not-allowed-by-RFC
- unknown-scheme
- whitespace-expected

- LWS-expected
- invalid-<SIP-Version>-on-request-line
- invalid-<protocol-name>
- invalid-<protocol-version>
- invalid-<transport>
- no-SLASH-after-<protocol-name>
- no-SLASH-after-<protocol-version>
- header-parameter-expected
- invalid-ttl-parameter
- invalid-maddr-parameter
- invalid-received-parameter
- invalid-branch-parameter
- invalid-rport-parameter
- via-parameter-repeat
- <seq>-number-expected
- <method>-expected
- <method>-does-not-match-the-request-line
- <response-num>-expected
- <CSeq-num>-expected
- <Method>-expected-after-<CSeq-num>
- expires-header-repeated
- <delta-seconds>-expected
- invalid-max-forwards
- token-expected
- invalid-expires-parameter
- invalid-q-parameter
- <generic-param>-with-invalid-<gen-value>
- <m-type>-expected
- SLASH-expected-after-<m-type>
- <m-subtype>-expected
- <m-attribute>-expected-after-SEMI
- boundary-parameter-appears-more-than-once
- EQUAL-expected-after-<m-attribute>
- invalid-<quoted-string>-in-<m-value>
- invalid-<m-value>
- multipart-Content-Type-has-no-boundary
- digits-expected
- IN-expected
- IP-expected
- IP4-or-IP6-expected

- IPv4-or-IPv6-address-expected
- line-order-error
- z-line-not-allowed-on-media-level
- <time>-expected
- <typed-time>-expected
- r-line-not-allowed-on-media-level
- <repeat-interval>-expected
- <bwtype>-execpted
- colon-expected
- <bandwidth>-expected
- t-line-not-allowed-on-media-level
- invalid-<start-time>
- invalid-<stop-time>
- too-many-i-lines
- <text>-expected
- too-many-c-lines
- too-many-v-line
- v-line-not-allowed-on-media-level
- too-many-o-lines
- o-line-not-allowed-on-media-level
- <username>-exepcted
- <sess-id>-expected
- <sess-version>-expected
- too-many-s-lines
- s-line-not-allowed-on-media-level
- too-many-m-lines
- <media>-expected
- <integer>-expected
- <proto>-expected
- <token>-expected-in-<proto>-after-slash
- <fmt>-expected
- <att-field>-expected
- <att-value>-expected
- <payload-type>-expected-in-rtpmap
- <encoding-name>-expected-in-rtpmap
- slash-expected-after-<encoding-name>-in-rtpmap
- invalid-<clock-rate>-in-rtpmap
- invalid-<encoding-parameters>-in-rtpmap
- invalid-candidate-line
- sdp-candidate-line-before-m-line

- sip-Yahoo-candidate-invalid-protocol
- invalid-port-after-ip-address-in-candidate-line
- too-many-candidate-lines
- sdp-invalid-alt-line
- sdp-alt-line-before-m-line
- invalid-port-after-ip-address-in-alt-line
- sdp-rtcp-line-before-m-line
- invalid-port-in-rtcp-line
- too-many-rtcp-lines
- <callid>-expected
- <word>-expected
- invalid-tag-parameter
- no-tag-parameter
- sdp-v-o-s-t-lines-are-mandatory
- unknown-header
- end-of-line-error
- sip-udp-message-truncated
- missing-mandatory-field

Traffic log fields

The table below lists the fields defined in traffic log tables (type tlog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For traffic logs, this field can be: <ul style="list-style-type: none"> • accept • deny • start
dir_disp	varchar(255)	varchar(255)	The direction of the sessions. Org displays if a session is not a child session or the child session originated in the same direction as the master session. Reply displays if a different direction is taken from the master session.
tran_disp	varchar(255)	varchar(255)	The packet is source NAT translated or destination NAT translated. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • noop • snat • dnat
srcname	varchar(255)	varchar(255)	The source name or the IP address.
dstname	varchar(255)	varchar(255)	The destination name or IP address.
tran_ip	varchar(40)	varchar(40)	The translated IP in NAT mode. For transparent mode, it is "0.0.0.0".

Field	Type		Description
	PostgreSQL	MySQL	
tran_port	int default 0	smallint unsigned default 0	The translated port number in NAT mode. For transparent mode, it is zero (0).
proto	int default 0	smallint unsigned default 0	The protocol that applies to the session or packet. The protocol number in the packet header that identifies the next level protocol. Protocol numbers are assigned by the Internet Assigned Number Authority (IANA).
app_type	varchar(255)	varchar(255)	The application or program used. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • N/A • BitTorrent • eDonkey • Gnutella • KaZaa • Skype • WinNY • AIM • ICQ • MSN • YAHOO
duration	bigint default 0	int unsigned default 0	This represents the value in seconds.
rule	bigint default 0	int unsigned default 0	The rule number.
sent	bigint default 0	int unsigned default 0	The total number of bytes sent.
rcvd	bigint default 0	int unsigned default 0	The total number of bytes received.
sent_pkt	bigint default 0	int unsigned default 0	The total number of packets sent during the session.
rcvd_pkt	bigint default 0	int unsigned default 0	The total number of packets received during the session.
vpn	varchar(255)	varchar(255)	The name of the VPN tunnel used by the traffic.
SN	bigint default 0	int unsigned default 0	The serial number of the log message.
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always displays N/A.
wanopt_app_type	varchar(255)	varchar(255)	The type of WAN optimization that was used. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • web-cache • cifs • tcp • ftp • mapi • http
wan_in	bigint default 0	int unsigned default 0	This field always displays WAN in.
wan_out	bigint default 0	int unsigned default 0	This field always displays WAN out.
lan_in	bigint default 0	int unsigned default 0	This field always displays LAN in.
lan_out	bigint default 0	int unsigned default 0	This field always displays LAN out.

Field	Type		Description
	PostgreSQL	MySQL	
app	varchar(255)	varchar(255)	The type of application. On the FortiGate unit, you can look the application type up in <i>UTM > Application Control > Application List</i> , and then select the name that is in the field to go to more detailed information on the FortiGuard Encyclopedia.
app_cat	varchar(255)	varchar(255)	The application category that the application is associated with.
shaper_drop_sent	bigint default 0	int unsigned default 0	The number of sent traffic shaper bytes that were dropped.
shaper_drop_rcvd	bigint default 0	int unsigned default 0	The number of received traffic shaper bytes that were dropped.
perip_drop	bigint default 0	int unsigned default 0	The number of per-IP traffic shaper bytes that were dropped.
shaper_sent_name	varchar(255)	varchar(255)	The name of the traffic shaper sending the bytes.
shaper_rcvd_name	varchar(255)	varchar(255)	The name of the traffic shaper receiving the bytes
perip_name	varchar(255)	varchar(255)	The name of the per-IP traffic shaper.

Antivirus log fields

The table below lists the fields defined in antivirus log tables (type vlog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For antivirus logs, this field can be: <ul style="list-style-type: none"> • blocked • passthrough • monitored
msg	varchar(255)	varchar(255)	Explains the activity or event that the FortiGate unit recorded. For example, the file that was downloaded from the web site exceeded the specified size limit.
sport	int default 0	smallint unsigned default 0	The source port of where the traffic is originating from.
dport	int default 0	smallint unsigned default 0	The destination port of where the traffic is going to.
serial	bigint default 0	int unsigned default 0	The serial number of the log message.
dir	varchar(255)	varchar(255)	Direction
filefilter	varchar(255)	varchar(255)	The file filter. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • none • file pattern • file type

Field	Type		Description
	PostgreSQL	MySQL	
filetype	varchar(255)	varchar(255)	<p>The file type. This field is an enum, and can have one of the following values:</p> <ul style="list-style-type: none"> • arj • cab • lzh • rar • tar • zip • bzip • gzip • bzip2 • bat • msc • uue • mime • base64 • binhex • com • elf • exe • hta • html • jad • class • cod • javascript • msoffice • fsg • upx • petite • aspack • prc • sis • hlp • activemime • jpeg • gif • tiff • png • bmp • ignored • unknown
file	varchar(255)	varchar(255)	The file name.
checksum	varchar(255)	varchar(255)	The file checksum.
quarskip	varchar(255)	varchar(255)	<p>This field is an enum, and can have one of the following values:</p> <ul style="list-style-type: none"> • No skip • No quarantine for HTTP GET file pattern block. • No quarantine for oversized files. • File was not quarantined.
virus	varchar(255)	varchar(255)	The virus name.
ref	varchar(255)	varchar(255)	The URL reference that gives more information about the virus. If you enter the URL in your web browser's address bar, the URL directs you to the specific page that contains information about the virus.

Field	Type		Description
	PostgreSQL	MySQL	
url	varchar(255)	varchar(255)	The URL address of where the file was acquired.
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always displays N/A.
agent	varchar(255)	varchar(255)	This field is for FortiGate units running FortiOS Carrier. If you do not have FortiOS Carrier running on your FortiGate unit, this field always displays N/A.
from	varchar(255)	varchar(255)	The from email address.
to	varchar(255)	varchar(255)	The to email address.
command	varchar(255)	varchar(255)	Protocol specific command, such as "POST" and "GET" for HTTP, "MODE" and "REST" for FTP.
dtype	varchar(255)	varchar(255)	Detection type, possible values: <ul style="list-style-type: none"> • virus • grayware

Web filter log fields

The table below lists the fields defined in web filter log tables (type wlog).

Field	Type		Description
	PostgreSQL	MySQL	
status	varchar(255)	varchar(255)	The status of the action the FortiGate unit took when the event occurred. For web filter logs, this field can be: <ul style="list-style-type: none"> • blocked • exempted • allowed • passthrough • filtered • DLP
serial	bigint default 0	int unsigned default 0	The serial number of the log message.
sport	int default 0	smallint unsigned default 0	The source port.
dport	int default 0	smallint unsigned default 0	The destination port.
hostname	varchar(255)	varchar(255)	The host name or IP.
carrier_ep	varchar(255)	varchar(255)	The FortiOS Carrier end-point identification. For example, it would display the MSISDN of the phone that sent the MMS message. If you do not have FortiOS Carrier, this field always displays N/A.
req_type	varchar(255)	varchar(255)	The request type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> • direct • referral
url	varchar(255)	varchar(255)	The URL.
msg	varchar(255)	varchar(255)	A text message explaining the log entry. For example, 'Message was blocked because it contained a banned word.'
dir	varchar(255)	varchar(255)	The direction.
agent	varchar(255)	varchar(255)	This field is for FortiGate units running FortiOS Carrier. If you do not have FortiOS Carrier running on your FortiGate unit, this field always displays N/A.

Field	Type		Description
	PostgreSQL	MySQL	
from	varchar(255)	varchar(255)	From
to	varchar(255)	varchar(255)	To
banword	varchar(255)	varchar(255)	The name of the banned word policy that triggered the log event.
error	varchar(255)	varchar(255)	The webfilter error.
method	varchar(255)	varchar(255)	The HTTP method. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> ip domain
class	smallint default 0	tinyint unsigned default 0	Class
class_desc	varchar(255)	varchar(255)	Class description
cat	smallint default 0	tinyint unsigned default 0	Category
cat_desc	varchar(255)	varchar(255)	Category description
mode	varchar(255)	varchar(255)	The mode. Can be 'rule' or 'off-site'.
rule_type	varchar(255)	varchar(255)	Rule type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> directory domain rating
rule_data	varchar(255)	varchar(255)	Rule data
ovrd_tbl	varchar(255)	varchar(255)	Override table
ovrd_id	bigint default 0	int unsigned default 0	Override ID
count	bigint default 0	int unsigned default 0	The number of scripts blocked by the scriptfilter within the page.
url_type	varchar(255)	varchar(255)	URL Type. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> http https ftp telnet mail
urlfilter_idx	bigint default 0	int unsigned default 0	URL Filter Index
urlfilter_list	varchar(255)	varchar(255)	URL Filter List
quota_exceeded	varchar(255)	varchar(255)	Quota Exceeded. Can be 'yes' or 'no'.
quota_used	bigint default 0	int unsigned default 0	Quota time used (in seconds).
quota_max	bigint default 0	int unsigned default 0	Maximum quota time allowed (in seconds).

Netscan log fields

The table below lists the fields defined in vulnerability / netscan log tables (type nlog).

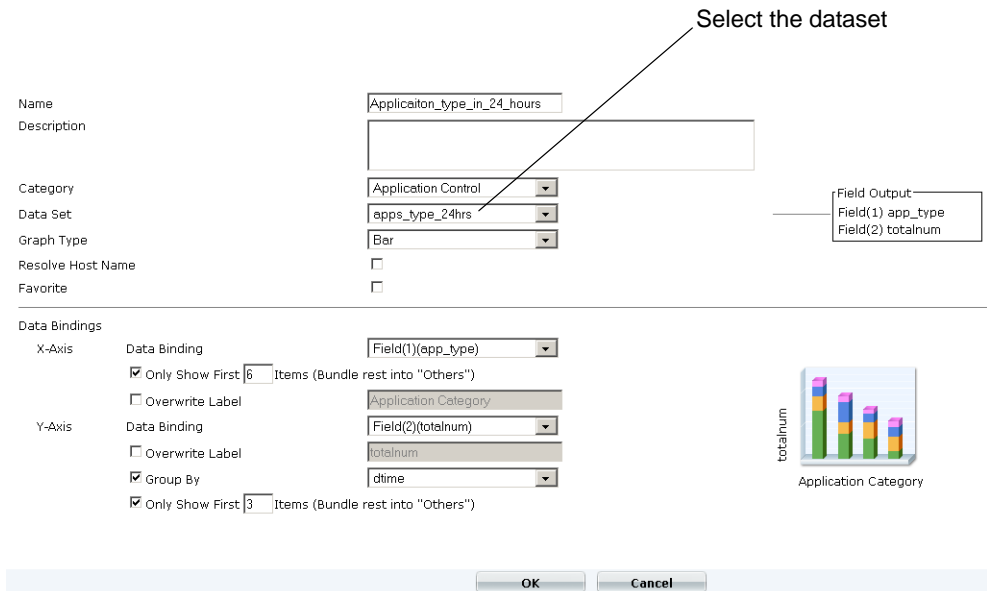
Field	Type		Description
	PostgreSQL	MySQL	
action	varchar(255)	varchar(255)	The nature of the event. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> scan vuln-detection host-detection service-detection
start	bigint default 0	int unsigned default 0	GMT epoch time the scan was started.
end	bigint default 0	int unsigned default 0	GMT epoch time the scan was started
engine	varchar(255)	varchar(255)	The netscan engine version.
plugin	varchar(255)	varchar(255)	The version of netscan plugins.
ip	varchar(40)	varchar(40)	The IP of the scanned asset.
proto	varchar(255)	varchar(255)	The protocol. Can be: <ul style="list-style-type: none"> tcp udp
port	int default 0	smallint unsigned default 0	The port scanned.
vuln	varchar(255)	varchar(255)	The name of the vulnerability found.
vuln_cat	varchar(255)	varchar(255)	The found vulnerability category.
vuln_id	bigint default 0	int unsigned default 0	The found vulnerability ID.
vuln_ref	varchar(255)	varchar(255)	A link to the detected vulnerability in FortiGuard.
severity	varchar(255)	varchar(255)	The severity of the vulnerability. This field is an enum, and can have one of the following values: <ul style="list-style-type: none"> critical high medium low info
os	varchar(255)	varchar(255)	The operating system of the scanned asset.
os_family	varchar(255)	varchar(255)	The family of the operating system on the scanned asset.
os_gen	varchar(255)	varchar(255)	The generation of the operating system on the scanned asset.
os_vendor	varchar(255)	varchar(255)	The vendor of the operating system on the scanned asset.
message	varchar(255)	varchar(255)	Informational message.

Examples

The following examples illustrate how to write custom datasets.

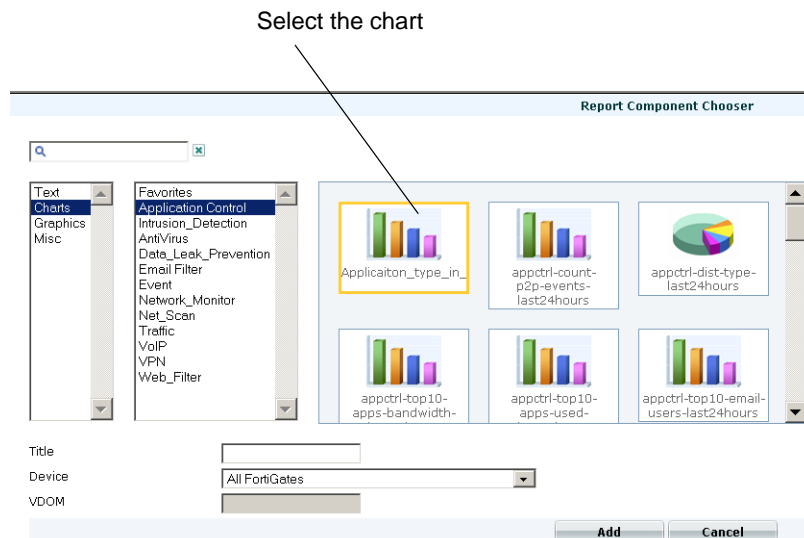
After you create the datasets, you can use them when you configure chart templates under *Report > Chart > Template*.

Figure 2: Adding a dataset to a chart template



Then you can use add the chart template to a report when you create the new report under *Report > Config > Report*.

Figure 3: Adding a chart to a report





Note: On the FortiGate unit, custom datasets can only be created via the CLI. On the FortiAnalyzer unit, datasets can be created via the CLI or the GUI. As well, on the FortiAnalyzer unit, queries support additional variables for log types (\$log) and time periods (\$filter) that make authoring queries easier.

Example 1: Distribution of applications by type in the last 24 hours

Figure 4: Creating a dataset

Name	apps_type_24hrs
Log Type(\$log)	Application Control
Time Period	Past N Hours
Past N Hours	24
SQL Query	<pre>AND app_type IS NOT NULL GROUP BY app_type ORDER BY totalnum DESC</pre>

Test

OK Cancel

GUI procedure

- 1 Go to *Report > Chart > Data Set*.
- 2 Click *Create New* to create a new dataset and enter a name (such as "apps_type_24hrs").
- 3 Under *Log Type(\$log)*, select *Application Control*.
- 4 Under *Time Period*, select *Past N Hours*, and enter 24 in *Past N Hours*.
- 5 Enter the query:

```
SELECT app_type, COUNT( * ) AS totalnum
FROM $log
WHERE $filter
AND app_type IS NOT NULL
GROUP BY app_type
ORDER BY totalnum DESC
```

CLI procedure

To perform the same task using the CLI, use these commands:

```
config sql-report dataset
  edit apps_type_24hrs
    set log-type app-ctrl
    set time-period last-n-hours
    set period-last-n 24
    set query "SELECT app_type, COUNT( * ) AS totalnum FROM $log
              WHERE $filter AND app_type IS NOT NULL GROUP BY app_type
              ORDER BY totalnum DESC"
  end
```

Notes:

- \$log queries all application control logs.

- \$filter restricts the query result to the time period specified; in this case, it's the past 24 hours.
- The application control module classifies each firewall session in app_type. One firewall session may be classified to multiple app_types. For example, an HTTP session can be classified to: HTTP, Facebook, etc.
- Some app/app_types may not be able to detected, then the 'app_type' field may be null or 'N/A'. These will be ignored by this query.
- The result is ordered by the total session number of the same app_type. The most frequent app_types will appear first.

Example 2: Top 100 applications by bandwidth in the last 24 hours

GUI procedure

- 1 Go to *Report > Chart > Data Set*.
- 2 Click *Create New* to create a new dataset and enter a name (such as "top_100_apps_24hrs").
- 3 Under *Log Type(\$log)*, select *Traffic*.
- 4 Under *Time Period*, select *Past N Hours*, and enter 24 in *Past N Hours*.
- 5 Enter the query:

```
SELECT (
TIMESTAMP - TIMESTAMP %3600
) AS hourstamp, app, service, SUM( sent + rcvd ) AS volume
FROM $log
WHERE $filter and app IS NOT NULL
GROUP BY app
ORDER BY volume DESC
LIMIT 100
```

CLI procedure

To perform the same task using the CLI, use these commands:

```
config sql-report dataset
  edit top_100_apps_24hrs
    set log-type traffic
    set time-period last-n-hours
    set period-last-n 24
    set query "SELECT ( TIMESTAMP - TIMESTAMP %3600 ) AS
hourstamp, app, service, SUM( sent + rcvd ) AS volume
FROM $log WHERE $filter and app IS NOT NULL GROUP BY app
ORDER BY volume DESC LIMIT 100"
  end
```

Notes:

- (timestamp-timestamp%3600) as hourstamp - this calculates an "hourstamp" to indicate bandwidth per hour.
- SUM(sent + rcvd) AS volume - this calculates the total sent and received bytes.

- ORDER BY volume DESC - this orders the results by descending volume (largest volume first)
- LIMIT 100 - this lists only the top 100 applications.

Example 3: Top 10 attacks in the past one hour

GUI procedure

- 1 Go to *Report > Chart > Data Set*.
- 2 Click *Create New* to create a new dataset and enter a name (such as "top_attacks_1hr").
- 3 Under *Log Type(\$log)*, select *Attack*.
- 4 Under *Time Period*, select *Past N Hours*, and enter 1 in *Past N Hours*.
- 5 Enter the query:

```
SELECT attack_id, COUNT( * ) AS totalnum
FROM $log
WHERE $filter and attack_id IS NOT NULL
GROUP BY attack_id
ORDER BY totalnum DESC
LIMIT 10
```

CLI procedure

To perform the same task using the CLI, use these commands:

```
config sql-report dataset
  edit top_attacks_1hr
    set log-type attack
    set time-period last-n-hours
    set period-last-n 1
    set query "SELECT attack_id, COUNT( * ) AS totalnum FROM
      $log WHERE $filter and attack_id IS NOT NULL GROUP BY
      attack_id ORDER BY totalnum DESC LIMIT 10"
  end
```

Notes:

- The result is ordered by the total attack number of the same attack_id. The most frequent attack_id will appear first.
- In a graph or report, the attack_id can be translated into the attack name.

Example 4: Top WAN optimization applications in the past 24 hours

GUI procedure

- 1 Go to *Report > Chart > Data Set*.
- 2 Click *Create New* to create a new dataset and enter a dataset name (such as "WAN_OPT_24hrs").
- 3 Under *Log Type(\$log)*, select *Traffic*.
- 4 Under *Time Period*, select *Past N Hours*, and enter 24 in *Past N Hours*.

5 Enter the query:

```
SELECT wanopt_app_type, SUM( wan_in + wan_out ) AS bandwidth
FROM $log
WHERE $filter
AND subtype = 'wanopt-traffic'
GROUP BY wanopt_app_type
ORDER BY SUM( wan_in + wan_out ) DESC
LIMIT 5
```

CLI procedure

To perform the same task using the CLI, use these commands:

```
config sql-report dataset
  edit WAN_OPT_24hrs
    set log-type traffic
    set time-period last-n-hours
    set period-last-n 24
    set query "SELECT wanopt_app_type, SUM( wan_in + wan_out )
      AS bandwidth FROM $log WHERE $filter AND subtype =
      'wanopt-traffic' GROUP BY wanopt_app_type ORDER BY SUM(
      wan_in + wan_out ) DESC LIMIT 5"
  end
```

Notes:

- The WAN optimizer module will log each application bandwidth. All bandwidth data is logged in traffic logs and wan opt data will have the subtype 'wanopt-traffic'
- SUM(wan_in + wan_out) AS bandwidth - this calculates the total in and out traffic.

Querying FortiGate SQL log databases

FortiGate units with hard disks support local SQLite databases for storage of log tables. You can enable logging to the local SQL database by going to *Log&Report > Log Config > Log Settings* and enabling SQL logging of different log types. Each log type has its own table in the database.

To create a report based on the FortiGate log messages in a database, you can use either the predefined datasets, or create your own custom datasets by querying the SQL database. The datasets can then be used in report charts, which are in turn used in log reports.

This document describes the procedures for displaying the SQL schema and creating custom datasets, as well as some example queries.

This section contains the following topics:

- [Creating datasets](#)
- [SQL tables](#)
- [Examples](#)

Creating datasets

The syntax for SQL queries is based on the SQLite3 syntax (see <http://www.sqlite.org/lang.html> for more information).

There is an additional convenience macro, `F_TIMESTAMP`, that allows you to easily specify a time interval for the query. It takes this form:

`F_TIMESTAMP(base_timestring, unit, relative value)`. For example, `F_TIMESTAMP('now', 'hour', '-23')` means “last 24 hours”, i.e. the hour in the timestamp is 23 less than now. The FortiGate unit will automatically translate the macro into SQLite3 syntax.

You can use the following CLI commands to write SQL statements to query the SQLite database.



Note: On FortiGate units, this feature is available in the CLI only.

```
config report dataset
  edit <dataset_name>
    set query <sql_statement>
  next
end
```

See the “[Examples](#)” section for specific examples of creating custom datasets.

SQL tables

The FortiGate unit creates a database table for each managed device and each log type, when there is log data. If the FortiGate unit is not receiving data from a device, or logging is not enabled under *Log&Report > Log Config > Log Settings*, it does not create log tables for that device.

The names used for SQL tables for each corresponding log type are listed in [Table 8 on page 56](#).

To view all the tables, column names, and column types, you can use this CLI command:

```
get report database schema
```

Table 8: Log types and table names

Log Type	SQL table name	Description
Traffic log	traffic_log	The traffic log records all traffic to and through the FortiGate interface.
Event log	event_log	The event log records management and activity events. For example, when an administrator logs in or logs out of the web-based manager.
Antivirus log	antivirus_log	The antivirus log records virus incidents in Web, FTP, and email traffic.
Webfilter log	webfilter_log	The web filter log records HTTP FortiGate log rating errors including web content blocking actions that the FortiGate unit performs.
Attack log	attack_log	The attack log records attacks that are detected and prevented by the FortiGate unit.
Spamfilter log	spamfilter_log	The spam filter log records blocking of email address patterns and content in SMTP, IMAP, and POP3 traffic.
Data Leak Prevention log	dlp_log	The Data Leak Prevention log records log data that is considered sensitive and that should not be made public. This log also records data that a company does not want entering their network.
Application Control log	app_control_log	The application control log records data detected by the FortiGate unit and the action taken against the network traffic depending on the application that is generating the traffic, for example, instant messaging software, such as MSN Messenger.
Vulnerability Management log	netscan_log	The vulnerability management log, or netscan log, contains logging events generated by a network scan.

FortiGate logs also include log subtypes, which are types of log messages that are within the main log type. For example, in the event log type there are the subtype admin log messages. FortiAnalyzer™ and FortiGate™ log types and subtypes are numbered, and these numbers appear within the log identification field of the log message.

Table 9: Log types and subtypes

Log Type	Sub-Type
traffic (Traffic Log)	<ul style="list-style-type: none"> • allowed – Policy allowed traffic • violation – Policy violation traffic • Other

Table 9: (Continued)Log types and subtypes

event (Event Log)	<p>For FortiGate devices:</p> <ul style="list-style-type: none"> • system – System activity event • ipsec – IPsec negotiation event • dhcp – DHCP service event • ppp – L2TP/PPTP/PPPoE service event • admin – admin event • ha – HA activity event • auth – Firewall authentication event • pattern – Pattern update event • alertemail – Alert email notifications • chassis – FortiGate-4000 and FortiGate-5000 series chassis event • sslvpn-user – SSL VPN user event • sslvpn-admin – SSL VPN administration event • sslvpn-session – SSL VPN session even • his-performance – performance statistics • vipssl – VIP SSL events • ldb-monitor – LDB monitor events
dlp (Data Leak Prevention)	<ul style="list-style-type: none"> • dlp – Data Leak Prevention
app-crtl (Application Control Log)	<ul style="list-style-type: none"> • app-crtl-all – All application control
DLP archive (DLP Archive Log)	<ul style="list-style-type: none"> • HTTP – Virus infected • FTP – FTP content metadata • SMTP – SMTP content metadata • POP3 – POP3 content metadata • IMAP – IMAP content metadata
virus (Antivirus Log)	<ul style="list-style-type: none"> • infected – Virus infected • filename – Filename blocked • oversize – File oversized
webfilter (Web Filter Log)	<ul style="list-style-type: none"> • content – content block • urlfilter – URL filter • FortiGuard block • FortiGuard allowed • FortiGuard error • ActiveX script filter • Cookie script filter • Applet script filter
ips (Attack Log)	<ul style="list-style-type: none"> • signature – Attack signature • anomaly – Attack anomaly
emailfilter (Spam Filter Log)	<ul style="list-style-type: none"> • SMTP • POP3 • IMAP

Log severity levels

You can define what severity level the FortiGate unit records logs at when configuring the logging location. The FortiGate unit logs all message at and above the logging severity level you select. For example, if you select Error, the unit logs Error, Critical, Alert, and Emergency level messages.

Table 10: Log severity levels

Levels	Description	Generated by
0 - Emergency	The system has become unstable.	Event logs, specifically administrative events, can generate an emergency severity level.
1 - Alert	Immediate action is required.	Attack logs are the only logs that generate an Alert severity level.
2 - Critical	Functionality is affected.	Event, Antivirus, and Spam filter logs.
3 - Error	An error condition exists and functionality could be affected.	Event and Spam filter logs.
4 - Warning	Functionality could be affected.	Event and Antivirus logs.
5 - Notification	Information about normal events.	Traffic and Web Filter logs.
6 - Information	General information about system operations.	Content Archive, Event, and Spam filter logs.

The Debug severity level, not shown in [Table 10](#), is rarely used. It is the lowest log severity level and usually contains some firmware status information that is useful when the FortiGate unit is not functioning properly. Debug log messages are only generated if the log severity level is set to Debug. Debug log messages are generated by all types of FortiGate features.

Examples

The following examples illustrate how to write custom datasets.

After you create the datasets, you can use them when you configure charts under *Log&Report > Report Config > Chart*.

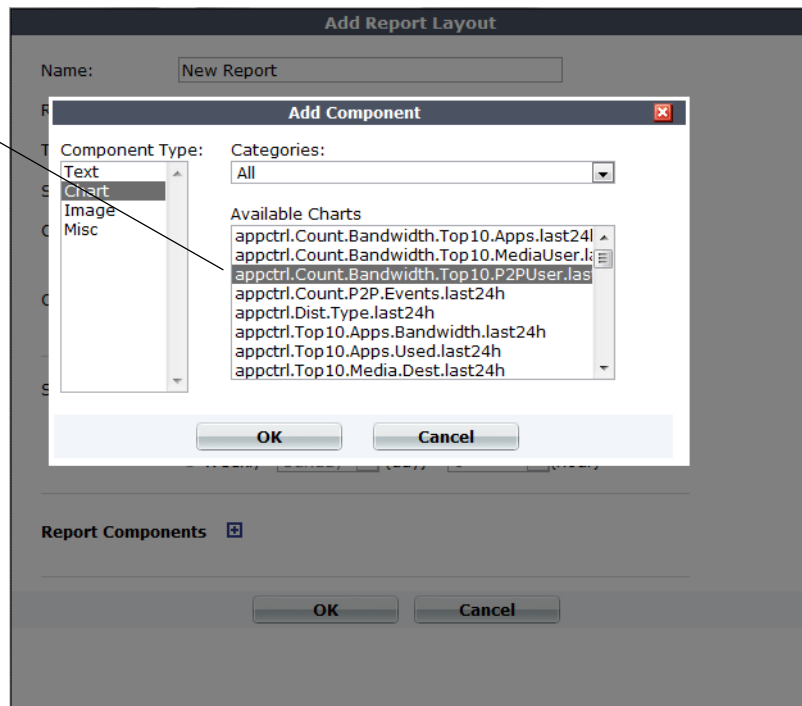
Figure 5: Adding a dataset to a chart

The screenshot shows the 'Edit Graph Report Chart' configuration window. The 'Dataset' dropdown menu is highlighted with a yellow border and a callout arrow pointing to it with the text 'Select the dataset'. The 'Name' field contains 'appctrl.Dist.Type.last24h'. The 'Category' is set to 'Application Control'. The 'Comments' field contains 'Distribution Of Apps By Type In Last 24 Hours'. The 'Graph Type' is set to 'Pie'. The 'Category Series' 'Databind' is set to 'app_type' and the 'Value Series' 'Databind' is set to 'totalnum'. There are 'OK' and 'Cancel' buttons at the bottom.

Then you can add the chart to a report when you create the new report under *Log&Report > Report Config > Layout*.

Figure 6: Adding a chart to a report

Select the chart



Note: On the FortiGate unit, custom datasets can only be created via the CLI. On the FortiAnalyzer unit, datasets can be created via the CLI or the GUI. As well, on the FortiAnalyzer unit, queries support additional variables for log types (\$log) and time periods (\$filter) that make authoring queries easier.

Example 1: Distribution of Applications by Type in the last 24 hours

CLI commands

```
config report dataset
edit "appctrl.Dist.Type.last24h"
set query "select app_type, count(*) as totalnum from
app_control_log where timestamp >=
F_TIMESTAMP('now', 'hour', '-23') and (app_type is not null
and app_type!='N/A') group by app_type order by totalnum
desc"
next
```

Notes:

- edit "appctrl.Dist.Type.last24h" - creates a new dataset with descriptive title.
- F_TIMESTAMP('now', 'hour', '-23') - the F_TIMESTAMP macro covers the last 24 hours (from now until 23 hours ago).
- The application control module classifies each firewall session in app_type. One firewall session may be classified to multiple app_types. For example, an HTTP session can be classified to: HTTP, Facebook, etc.
- Some app/app_types may not be able to detected, then the 'app_type' field may be null or 'N/A'. These will be ignored by this query.

- The result is ordered by the total session number of the same app_type. The most frequent app_types will appear first.

Example 2: Top 10 Application Bandwidth Usage Per Hour Summary

CLI commands

```
config report dataset
edit "appctrl.Count.Bandwidth.Top10.Apps.last24h"
set query "select (timestamp-timestamp%3600) as hourstamp,
(CASE WHEN app!=\"N/A\" and app!=\"\" then app ELSE service
END) as appname, sum(sent+rcvd) as bandwidth from
traffic_log where timestamp >=
F_TIMESTAMP(\"now\", \"hour\", \"-23\") and (appname in
(select (CASE WHEN app!=\"N/A\" and app!=\"\" then app ELSE
service END) as appname from traffic_log where timestamp >=
F_TIMESTAMP(\"now\", \"hour\", \"-23\") group by appname
order by sum(sent+rcvd) desc limit 10)) group by hourstamp,
appname order by hourstamp desc"
next
```

Notes:

- (timestamp-timestamp%3600) as hourstamp - this calculates an "hourstamp" to indicate bandwidth per hour.
- (CASE WHEN app!=\"N/A\" and app!=\"\" then app ELSE service END) as appname - use the app as 'appname', or if it's undefined, use the service instead.
- appname in (select (CASE WHEN app!=\"N/A\" and app!=\"\" then app ELSE service END) as appname from traffic_log where timestamp >= F_TIMESTAMP(\"now\", \"hour\", \"-23\") group by appname order by sum(sent+rcvd) desc limit 10) - selects the top 10 apps using most bandwidth
- order by hourstamp desc - this orders the results by descending hourstamp
- LIMIT 10 - this lists only the top 10 applications.

Example 3: Top 10 Attacks Over The Last 24 Hours

CLI commands

```
config report dataset
edit "attack.Top10.last24h"
set query "select attack_id, count(*) as totalnum from
attack_log where timestamp >= F_TIMESTAMP('now', 'hour', '-
23') and attack_id is not null group by attack_id order by
totalnum desc limit 10"
next
```

Notes:

- The result is ordered by the total attack number of the same attack_id. The most frequent attack_id will appear first.
- In a graph or report, the attack_id can be translated into the attack name.

Example 4: Wan Optimization Application in LAN Composition over Last 24 Hours

CLI commands

```
config report dataset
  edit "traffic.Dist.WanOpt.App.WAN.Bandwidth.last24h"
    set query "select (case (wanopt_app_type in ( select
      wanopt_app_type from traffic_log where subtype='wanopt-
      traffic\' and timestamp >=
      F_TIMESTAMP('\now\',\'hour\',\'-23\') group by
      wanopt_app_type order by sum(wan_in+wan_out) desc limit 5)
    ) when 1 then wanopt_app_type else \'others\' end) as
      wanopt_app_type, sum(wan_in+wan_out)/1000000.0 as wan,
      max(coalesce((sum(wan_in+wan_out)*100.0/(select
      sum(wan_in+wan_out) from traffic_log where
      subtype='wanopt-traffic\' and timestamp >=
      F_TIMESTAMP('\now\',\'hour\',\'-23\'))),0.0),0.0) as
      percentage from traffic_log where subtype='wanopt-
      traffic\' and timestamp >=F_TIMESTAMP('\now\',\'hour\',\'-
      23\') group by wanopt_app_type order by wan desc"
  next
```

Notes:

- This is a very complex SQL statement.
- The WAN optimizer module will log each application's bandwidth. All bandwidth data is logged in traffic logs and wan opt data will have the subtype 'wanopt-traffic'.
- `select wanopt_app_type from traffic_log where subtype='wanopt-traffic\' and timestamp >= F_TIMESTAMP('\now\',\'hour\',\'-23\')` group by wanopt_app_type order by sum(wan_in+wan_out) desc limit 5 - find 5 wanopt_app_types who consume most of the bandwidth.
- `case (wanopt_app_type in (...)) when 1 then wanopt_app_type else \'others\' end) as wanopt_app_type` - select only the 5 wanopt_app_types who consume most of the bandwidth, keep the wanopt_app_type name, and all other wanopt_app_types are grouped as 'others'.
- `sum(wan_in+wan_out)/1000000.0 as wan` - calculate in and out traffic and convert to MB
- `max(coalesce((sum(wan_in+wan_out)*100.0/(select sum(wan_in+wan_out) from traffic_log where subtype='wanopt-traffic\' and timestamp >= F_TIMESTAMP('\now\',\'hour\',\'-23\'))),0.0),0.0) as percentage` - calculate (one wanopt_app_type traffic / all wanopt traffic) as percentage

FORTINET®

www.fortinet.com

FORTINET®

www.fortinet.com