



© Panda Adaptive Defense 360

Panda Advanced Reporting Tool Administration Guide

Version: 1.02.00-02

Author: Panda Security

Date: 05/11/2020



[pandasecurity.com](https://www.pandasecurity.com)

Legal notice.

Neither the documents nor the programs that you may access may be copied, reproduced, translated or transferred to any electronic or readable media without prior written permission from Panda Security, C/ Santiago de Compostela, 12, 48003 Bilbao (Bizkaia), SPAIN.

Registered trademarks.

Windows Vista and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. All other product names may be registered trademarks of their respective owners.

© Panda Security 2020. All rights reserved

Contact information.

Corporate Headquarters:

Panda Security

Calle Santiago de Compostela 12

Bilbao (Bizkaia) 48003 Spain.

<https://www.pandasecurity.com/uk/about/contact/>

About the Administration Guide

You can find the most recent version of this guide at:

<https://www.pandasecurity.com/rfiles/enterprise/solutions/adaptivedefense/ADVANCEDREPORTINGTOOL-Guide-EN.pdf>

Panda Adaptive Defense and Panda Adaptive Defense 360 guides

<http://www.pandasecurity.com/rfiles/enterprise/solutions/adaptivedefense/ADAPTIVEDEFENSE360-guide-2.40.0-EN.pdf>

<http://www.pandasecurity.com/rfiles/enterprise/solutions/adaptivedefense/ADAPTIVEDEFENSE-guide-2.40.0-EN.pdf>

Technical Support

Panda Security provides global support services aimed at responding to specific questions regarding the operation of the company's products. The technical support team also generates documentation covering technical aspects of our products. This documentation is available in the eKnowledge Base portal.

- To access specific information about the product, please go to the following URL:

<https://www.pandasecurity.com/en/support/advanced-reporting-tool/>

Survey on the Administration Guide

Rate this guide and send us suggestions and requests for future versions of our documentation:

<https://es.surveymonkey.com/r/feedbackARTGuideEN>

Contents

Part 1: Introduction to Panda Advanced Reporting Tool

Chapter 1: Preface	9
Who is this guide aimed at?	9
Icons	9
Chapter 2: Introduction	11
Main benefits	12
Main features of the Panda Advanced Reporting Tool service	12
Accumulated information	12
Main components of the Panda Advanced Reporting Tool architecture	13
Other additional services	16
Panda Advanced Reporting Tool user profile	16
Chapter 3: The Web console	17
Requirements for accessing the Web console	18
Accessing the Panda Advanced Reporting Tool Web console	18
Structure of the Web console	18
Side menu overview	19

Part 2: Panda Advanced Reporting Tool resources

Chapter 4: Introduction to the applications	23
Accessing the dashboards/applications	24
Resources and common dashboard items	24
Time periods for the data displayed	24
Tabs	25
Sections	25
Widgets	25
Search tool	26
Tables and charts	27
Pre-configured alerts	32
Accessing pre-configured alerts and setting the delivery frequency	32
Generating new charts based on the widgets provided	33
Modifying the SQL statement associated with a widget	33
SQL statement favorites	34
Chapter 5: Configured applications	35
Setting the time period	36
Associated alerts	37
Security Incidents application	38
Key Security Indicators	38
Detailed Information	39
Application Control application	41
IT Applications	41
Vulnerable Applications	43
Bandwidth-consuming Applications	45
Special Applications & Tools	46
Data Access Control Application	50
Outbound network traffic	50

User activity.....51
 Bandwidth consumers.....52
 Data Files Accessed53

Chapter 6: Alerts - - - - - 55
 Alert system architecture.....56
 Process for configuring the alerts.....56
 Creating alerts57
 Alert management59
 Creating post filters61
 Post filter management.....62
 Creating delivery conditions.....63
 Delivery method management66
 Creating antiflooding policies67
 Creating alert policies or delivery methods67
 Editing sending policies.....68
 Configuring an alert sending policy68

Part 3: Additional information

Chapter 7: Knowledge table - - - - - 73
 Terminology used in fields.....73
 Alert74
 Install80
 Monitoredopen81
 MonitoredRegistry83
 Notblocked84
 Ops85
 ProcessNetBytes.....88
 Registry.....92
 Socket93
 ToastBlocked99
 URLdownload.....100
 VulnerableAppsFound104

Chapter 8: Hardware, software and network requirements- - - - - 107
 Management console access requirements.....107
 Hardware requirements.....107



Part 1

Introduction to Panda Advanced Reporting Tool

Chapter 1: Preface

Chapter 2: Introduction

Chapter 3: The Web console

Chapter 1

Preface

This guide offers the information and procedures necessary to benefit fully from the Panda Advanced Reporting Tool service.

CHAPTER CONTENT

Who is this guide aimed at?	9
Icons	9

Who is this guide aimed at?

The documentation is aimed at technical personnel in IT departments of organizations that have contracted the Panda Advanced Reporting Tool service for Panda Adaptive Defense and Panda Adaptive Defense 360.

This manual includes the procedures and settings required to interpret and fully benefit from the security information provided by the Panda Advanced Reporting Tool platform.

All the procedures and instructions in this guide apply both to Panda Adaptive Defense and Panda Adaptive Defense 360. The term "Panda Adaptive Defense" is used generically to refer to both of these advanced security products.

Icons

The following icons are used in the guide;



Additional information, such as an alternative way of performing a certain task.



Suggestions and recommendations.



Important advice regarding the proper use of the options available in the Panda Advanced Reporting Tool service.



See another chapter or section in the guide for more information.

Chapter 2

Introduction

Panda Advanced Reporting Tool is an advanced, real-time service for leveraging the knowledge generated by Panda Adaptive Defense and Panda Adaptive Defense 360.

Its main aim is to enable the discovery of unknown threats, targeted attacks designed to steal confidential information from companies, and APTs (Advanced Persistent Threats). To achieve this, it represents the data relating to processes run by users, with particular emphasis on events related to security and the extraction of data from the organization's IT resources.

It can also determine what network users can do with their computers, both in terms of bandwidth usage by applications and the use of installed applications. It facilitates the identification of applications that have vulnerabilities which could be exploited by latest generation malware.

Panda Advanced Reporting Tool implements tools for performing advanced searches of the information repository and allows new configurations and representations of the stored data to be developed. These are flexible representations that adapt to the needs of technical personnel when generating intelligent security to detect malicious processes that would otherwise 'slip under the radar'.

When all resources are implemented, Panda Advanced Reporting Tool is the most complete tool for accurately determining the network security status.

CHAPTER CONTENT

Main benefits	12
Main features of the Panda Advanced Reporting Tool service	12
Accumulated information	12
Main components of the Panda Advanced Reporting Tool architecture	13
Cloud-hosted infrastructure	14
Panda Advanced Reporting Tool server	14
Computers protected by Panda Adaptive Defense and Panda Adaptive Defense server ...	15
Management console Web server and network administrator's computer	15
Applications / Dashboards	15
Accumulated knowledge tables	16
Other additional services	16
Panda Advanced Reporting Tool user profile	16

Main benefits

The main benefits of Panda Advanced Reporting Tool derive from the visualization of the activity of network processes to automatically generate security intelligence.

- It displays the progress of all types of malware detected on a customer's network, indicating whether or not it has been executed in order to facilitate remedial action and the adjustment of security policies.
- It lists the actions run by each process, whether goodware, malware or unknown, in order to compile data that can be used to reach conclusions about its potential risk.
- It enables visualization of attempts to access confidential information to prevent leakage or theft.
- It locates all executed programs, especially those with known vulnerabilities installed on the users' computers, in order to help design a plan for updating software.
- It helps to properly dimension available network resources, displaying those applications and users that require most network bandwidth.

Main features of the Panda Advanced Reporting Tool service

Panda Advanced Reporting Tool transforms the bulk data gathered by Panda Adaptive Defense into security intelligence with different levels of detail. To achieve this, it employs a series of tools and resources:

- A wide range of configurable graphic widgets to enable visualization of the activity data.
- Dashboards that can be configured by administrators with relevant information for the IT department.
- Configurable real-time alerts to identify potentially dangerous situations.
- Knowledge charts with detailed information about the actions provoked by all processes run on users' computers.
- Advanced search and processing tools for the stored data: filtering, grouping, advanced data processing, generation of new widgets with information, etc.

Accumulated information

The Panda Advanced Reporting Tool service stores the information generated in real time by the network computers with Panda Adaptive Defense.

Most of the information collected is generated as a result of the active monitoring of processes run on customers' computers. This monitoring is performed by Panda Adaptive Defense, and Panda Advanced Reporting Tool takes care of storing and organizing the data by type, as well as generating charts that enable the data to be interpreted.

Some of the events logged by Panda Adaptive Defense and displayed by 1 are as follows:

- Installation and uninstallation of drivers on the operating system.
- Installation and modification of keyboard, mouse and other device hooks.
- Modifications to the registries of Windows computers on the network.
- Modifications to the system file (`HOSTS`).
- Record of the volume of data sent and received by each process across the network.
- Record of communications established with remote systems.
- Software with known vulnerabilities installed on computers.
- Execution and termination of processes.
- Loading of libraries.
- Manipulation of the file system.
- Running of the command line.

The logged events could be related to the execution of unknown malicious code and as such Panda Advanced Reporting Tool is a fundamental tool for monitoring processes to identify suspicious behavior.

Main components of the Panda Advanced Reporting Tool architecture

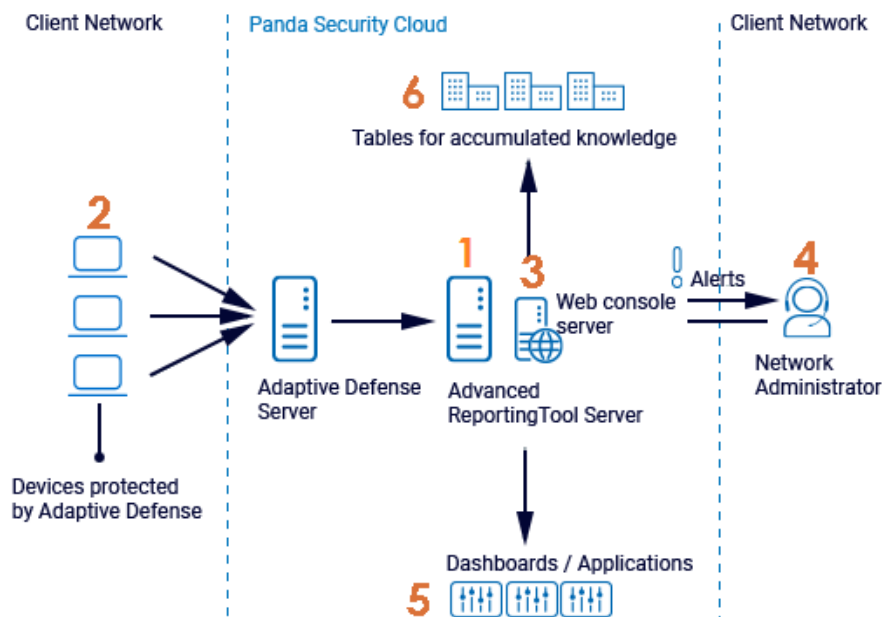


Figure 2.1: Panda Advanced Reporting Tool architecture

Panda Advanced Reporting Tool comprises the following components:

- Panda Advanced Reporting Tool server **(1)**.

- Computers protected by Panda Adaptive Defense or Panda Adaptive Defense 360 **(2)**.
- Web management console server **(3)**.
- Network administrator computer for managing the service **(4)**.
- Applications / Dashboards **(5)**.
- Stored data tables **(6)**.

Cloud-hosted infrastructure

All the infrastructure directly involved in the service (Panda Advanced Reporting Tool service, Panda Adaptive Defense server, Web console server) is deployed in the Panda Security cloud, with the following advantages:

- **No maintenance costs for the customer**

As the servers do not have to be physically installed on customers' premises, customers can forget about the costs arising from the purchasing and maintenance of hardware (warranty management, technical problems, storage of spare parts, etc.).

Neither will they have to worry about costs associated with operating systems, databases, licenses or other factors associated with on-premises solutions.

Similarly, the outlay derived from needing specialized personnel to maintain the solution also disappears.

- **Access to the service from anywhere at any time**

The service can be securely accessed from any computer on the customer's network, thereby countering the problems that occur in companies with an IT structure spread across several locations. For this reason, it is not necessary to have specific communication deployments, such as VPNs, or special router configurations to enable access to the management console from outside the customer's local network.

- **Service available 24/7 - 365 days a year**

This is a high availability service, with no limit on the number of monitored computers. Customers do not need to design or implement complex redundant infrastructure configurations. Nor do they require specific technical personnel to maintain service availability.

Panda Advanced Reporting Tool server

This is a high availability server farm that harvests all the events sent by the Panda Adaptive Defense agents installed on users' computers.

The sending and collection of data is continuous in real time. The server stores the data in tables that can be readily accessed by administrators, while generating straightforward graphic data and configurable alerts to advise of potentially dangerous situations.

Computers protected by Panda Adaptive Defense and Panda Adaptive Defense server

Users' computers continually send the actions executed by processes to the cloud-hosted Panda Adaptive Defense server. This server automatically generates security intelligence through Machine Learning technologies on Big Data repositories. The security intelligence is added to the events collected from the computers protected by Panda Adaptive Defense and are sent directly to the Panda Advanced Reporting Tool server. This operational structure offers the following advantages:

- The information received by the Panda Advanced Reporting Tool server is already processed by the Panda Adaptive Defense server, and as such contains the security intelligence that will help identify problems caused by malware.
- Data packets are only sent once from the protected computers protected by Panda Adaptive Defense, saving bandwidth and the need to install SIEM servers locally in every location, which would be much more complex and expensive to maintain.
- No additional configuration is required, neither in the Panda Adaptive Defense console, nor on the protected computers. The Panda Adaptive Defense servers will automatically and transparently send all necessary information to the Panda Advanced Reporting Tool server.

Management console Web server and network administrator's computer

The Web server hosts the management console, accessible from any place at any time through any ordinary compatible browser.



See [“The Web console”](#) on page 17 for more information.

Applications / Dashboards

The most relevant information for the IT team is displayed through three applications accessible from the Web management console:

- **Security Incidents:** This lets you view malware activity across the organization.
- **Application Control:** This displays information about the applications installed across the network.
- **Data Access Control:** This shows the information accessed by users as well as bandwidth usage.

All the applications are interactive and allow more detailed information to be obtained by clicking on the displayed items.



See [“Introduction to the applications”](#) on page 23 for more information about the applications.

Accumulated knowledge tables

The system stores the data received by the Panda Adaptive Defense server in 15 tables which can be easily accessed by the IT department.

These tables are used as the source for generating the charts and allow numerous types of filtering and other actions (grouping data, organizing the information, searches, etc.).



See "[Knowledge table](#)" on page [73](#) for more information about the accumulated knowledge tables and the meaning of each field

Other additional services

With the purchase of the SIEM Feeder service, the network administrator will be able to incorporate all the information generated by the activity of the processes run on their IT network into the company's SIEM solution. Moreover, this information is enriched with the security intelligence developed by Panda Security.

The information processed by Panda Advanced Reporting Tool and documented in chapter 7 is a subset of the volumes of data that Panda Security makes available to customers for exploitation via SIEM Feeder.



For more information about Panda SIEMFeeder and the data sent to the customer's server, refer to the [Panda SIEMFeeder Administrator's Guide](#)

Panda Advanced Reporting Tool user profile

This service is primarily aimed at the IT department of organizations, who can carry out some or all of the tasks below:

- Monitoring the activity of processes run on users' computers.
- Monitoring the general security status of the network.
- Developing policies to protect the organization's data and confidential information.
- Generating data for forensic analysis in the event of malware infections.
- Generating additional information for auditing computers.
- Dimensioning the bandwidth required for the organization's activities.
- Generating additional information for security audits.

Chapter 3

The Web console

This chapter describes the general structure of the Web management console and its components.

The Web console is the main tool for administrators to view the security status of the network. As a centralized Web service, it offers a series of features that positively affect the way the IT department can work with it:

- **A single tool for leveraging security information**

The Web console allows you to monitor the security status of the network and provides pre-configured tools to represent and interpret all the collected information.

All of this is delivered via a single Web console, enabling the integration of various tools and removing the complexity of using products from different vendors.

- **Access to consolidated information without the need to support infrastructure across all locations**

As the server that hosts the Web console is hosted by Panda Security there is no need to install or maintain specific infrastructure on customers' premises.

Moreover, as it is hosted in the cloud, the server can be accessed from all customers' offices, presenting consolidated data from a single repository. This simplifies data interpretation and speeds up decision making.

CHAPTER CONTENT

Requirements for accessing the Web console	18
Accessing the Panda Advanced Reporting Tool Web console	18
Structure of the Web console	18
Side menu overview	19
Home	19
Search	19
Administration	19
Applications	20
Alerts	20
Preferences	20
Log out	20

Requirements for accessing the Web console

To access the Panda Advanced Reporting Tool Web console, the following requirements should be taken into account:

- A certified compatible browser (other browsers may work)
 - Mozilla Firefox
 - Google Chrome



Other browsers may be compatible but not all versions are supported. As such it is advisable to use one of the browsers listed above

- Internet connection and communication through port 443
- Minimum screen resolution 1280x1024 (1920x1080 recommended)
- A sufficiently powerful computer to generate
 - charts and lists in real time
 - Sufficient bandwidth to display all the information collected from users' computers in real time

Accessing the Panda Advanced Reporting Tool Web console

The Panda Advanced Reporting Tool Web console can be accessed via SSO using the Panda Adaptive Defense management console, with no need to enter new credentials.

- Click on **Status** top menu

To access the Panda Advanced Reporting Tool environment, select the **Advanced search** option from the top menu in To access the Panda Advanced Reporting Tool environment, select the Panda Advanced Reporting Tool link from the top menu in Panda Adaptive Defense.



Figure 3.1: Access to Panda Advanced Reporting Tool from Panda Adaptive Defense console.

Structure of the Web console

The Web console is designed to deliver a uniform and coherent experience to administrators, both in terms of visualization and the search for information as well as configuring custom data panels.

The end goal is to deliver a simple yet powerful and flexible tool that allows administrators to rapidly assess the security status of the network without a steep learning curve.

Side menu overview

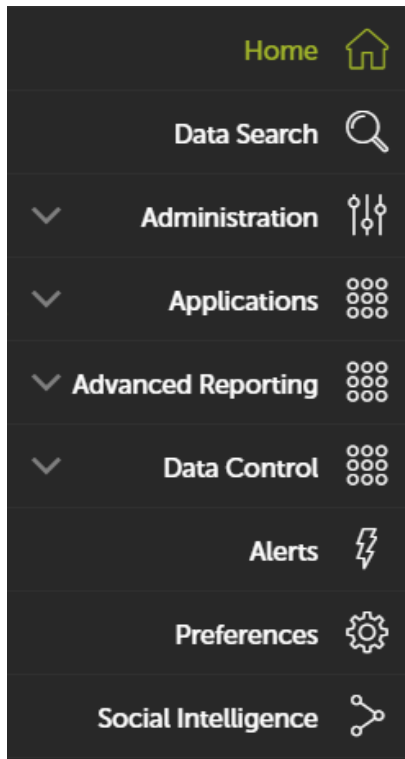


Figure 3.2: Side menu

The side menu is located to the left of the screen and can be accessed at any time.

Initially, this menu only displays the icons for each option. By moving the mouse pointer to the left of the screen, or clicking a free section of the side menu, a description of each icon is displayed

Below you can see the main options of the side menu:

Home


This takes users back to the Home page of the Web console.

Search

This lets you access the accumulate knowledge tables. From here, administrators can view the data as it has been sent from the computers protected by Panda Adaptive Defense.

As administrators access the knowledge tables, they appear under the Search option as shortcuts, to make it easier to access

them.



See "[Knowledge table](#)" on [page 73](#) for more information about the accumulated knowledge tables

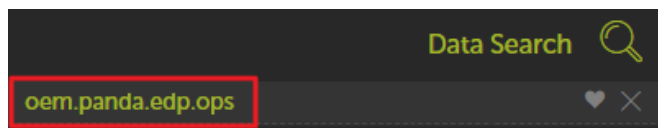



Figure 3.3: Direct access to the knowledge table

Administration

This lets you configure new alerts.



For more information about pre-configured alerts, see "[Alerts](#)" on [page 55](#). For more information about how to create and configure new alerts, see "[Creating alerts](#)" on [page 57](#)

Applications

The **Applications** menu has a drop-down menu with the applications available to the network administrator. The applications are interactive, pre-configured dashboards that process and present the data gathered in a simple and clear format. All the applications allow you to define the time period for the collection and presentation of data.

These include the three applications described below.

- **Security Incidents:** This displays the security status and the incidents detected on the network, along with information that lets you determine the source of threats and the impact on the organization.
- **Application Control:** This displays data regarding the use of the applications installed across the network.
- **Data Access Control:** This displays information about bandwidth usage and access to documents by the applications installed across the network.



For more information about applications, see "[Configured applications](#)" on page 35

Alerts

This displays a window with information about the alerts received.



For more information about pre-configured alerts, see "[Alerts](#)" on page 55. For more information about how to create and configure new alerts, see "[Creating alerts](#)" on page 57

Preferences

This section offers a series of options that can be configured for the logged-in user and for others that access the service.

Log out

Here you can log out of the Panda Advanced Reporting Tool console. It then displays the IDP (Identity Provider) login screen.



Part 2

Panda Advanced Reporting Tool resources

Chapter 4: Introduction to the applications

Chapter 5: Configured applications

Chapter 6: Alerts

Chapter 4

Introduction to the applications

The dashboards are pre-configured applications that provide the network administrator with specific information about the network.

The three dashboards included in the Web console are as follows:

- Security Incidents.
- Application Control.
- Data Access Control.

All the dashboards have a common layout, described later in "[Resources and common dashboard items](#)", in order to facilitate data interpretation.

The applications also generate alerts that warn administrators in real time of potential problems.



To create new alerts in addition to those that are already configured in the applications, see "[Creating alerts](#)" on page 57

CHAPTER CONTENTS

Accessing the dashboards/applications	-24
Accessing the dashboards/applications	24
Accessing the alerts	24
Resources and common dashboard items	-24
Time periods for the data displayed	24
Tabs	25
Sections	25
Widgets	25
Search tool	26
Tables and charts	27
Calendar charts	27
Stacked bar chart	27
World map chart	28
Voronoi diagram	28
Pre-configured alerts	-32
Accessing pre-configured alerts and setting the delivery frequency	32

Generating new charts based on the widgets provided	33
Modifying the SQL statement associated with a widget	33
SQL statement favorites	34

Accessing the dashboards/applications

Accessing the dashboards/applications

Access to the dashboards is available through the side menu, in the **Applications** section.

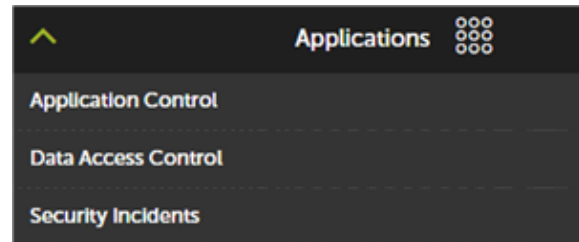


Figure 4.1: Applications drop-down menu

Accessing the alerts

Access to the alerts is available through the side menu, through **Administration, Alerts Configuration**.

The **Alerts Subscription** screen is used to look for configured alerts, to assign policies, and enable and disable individual alerts.

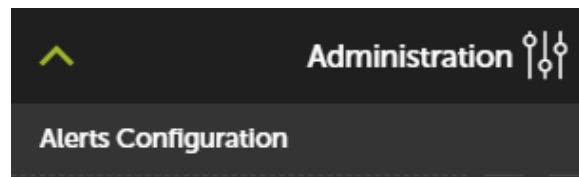


Figure 4.2: Administration menu entry for the configuration of existing alerts

Resources and common dashboard items

Time periods for the data displayed

Each application has two controls for defining the time period for the data displayed on screen:

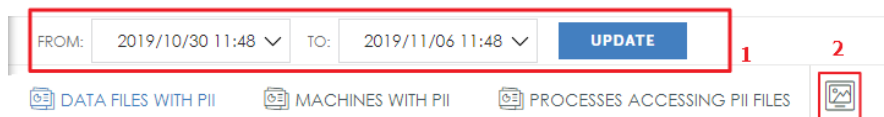


Figure 4.3: Controls to configure the ranges to display

- **Date range (1):** This lets you set the time period displayed in the widgets of the selected dashboard. The period will apply to the widgets of all the tabs on the dashboard.
- **Screenshot (2):** This opens an independent window with the content of the tab in graph format so it can be downloaded and printed.



The browser pop-up protection may prevent you from seeing the new window. Disable this feature in the browser in order to see the window

Tabs

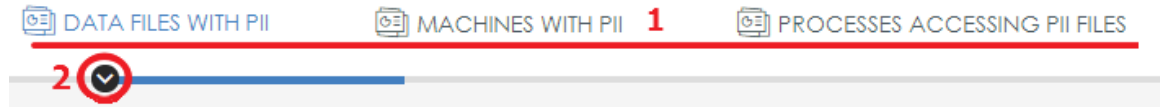


Figure 4.4: Console tabs

The tabs divide the information into different areas according to the level of detail of the data displayed: general information or more detailed reports and data breakdowns.

Each tab offers access to the tools displayed below:

- **Tab name (1):** This describes the information contained in the tab. To select a tab, simply click on the name. The **Detailed information** tabs contain data tables that can be used in reports.
- **Shortcut menu (2):** Click the arrow to display a drop-down menu that takes you directly to any section within the tab.

Sections

The information within a tab is divided into sections. Each section is a group of widgets with related information.

Click the arrow button to display or hide a complete section.



Figure 4.5: Accessing a tab's sections

Widgets

These are controls that display the data using tables and advanced graphs.

Incidents Type ¹ 2 ↓ ≡ 3

ALERTTYPE	COUNT	%
Malware	51	78.46%
PUP	12	18.46%
Exploit	2	3.08%

Figure 4.6: Widget

Each widget comprises the following items (some may be missing depending on the widget type):

- **Widget name (1):** This indicates the type of information displayed.
- **Display/hide button (2) ↓:** This lets you hide or display the widgets you want.
- **Widget menu (3) ≡:** This contains four options:
 - **Screenshot:** This opens the widget content on a new page so it can be saved as a graph, printed, etc.

- **Download data:** This downloads the raw data displayed in the widget. The data is downloaded in comma-separated (CSV) format to be imported into other applications.
- **Go to query:** This displays the knowledge table that serves as data source for the widget, along with the filter, grouping, and operation settings applied.



The **Go to query** menu displays the exact settings of the data source that feeds the widget, including the specified time interval. This allows the administrator to experiment with multiple variations of the chart displayed, taking as base the SQL statement used. See later in the chapter for more information.

- **Zoom:** This displays the widget in full-screen mode.

Search tool

Some table-type widgets incorporate a search tool that is very useful to find specific items within all contents of the tables - except in percentages and counters.

Tables contain a maximum of 1,000 records, sorted in descending order.

The search tool starts searching as soon as you start typing, as seen in the figure below, in which the user is looking for 'Malware' alerts:

Incidents on all endpoints ⌵ ☰

Buscar: ✕

ALERT TYPE	MACHINE NAME	EXECUTION STATUS	PROGRAM
Malware	WIN_LAPTOP_4	Executed	PROFILE \downloads\beyond_compare_3.1.1104_crack_downloader.exe
Malware	WIN_SERVER_3	Not Executed	TEMP \calc1.exe
Malware	WIN_DESKTOP_4	Executed	TEMP \23a2de88288f64c9a3e89a2e7eba3be7
Malware	WIN_LAPTOP_2	Executed	TEMP \62b2153392561255386e5f059c2161cd

Figure 4.7: Table of the data corresponding to "Malware"

Or, take this other example, in which the user is looking for programs located in the TEMP folder:

Incidents on all endpoints ⌵ ☰

Buscar: ✕

ALERT TYPE	MACHINE NAME	EXECUTION STATUS	PROGRAM	THREAT
Malware	WIN_SERVER_3	Not Executed	TEMP \calc1.exe	Trj/Chgt.J
Malware	WIN_LAPTOP_2	Executed	TEMP \62b2153392561255386e5f059c2161cd	Trj/WLT.B
Malware	WIN_DESKTOP_4	Executed	TEMP \23a2de88288f64c9a3e89a2e7eba3be7	Trj/Chgt.J

Figure 4.8: Programs in the TEMP directory



The browser pop-up protection may prevent you from seeing the new window. Disable this feature in the browser in order to see the window

Tables and charts

The data is represented through a range of charts (Voronoi, line and bar charts, pie charts, etc.) and more detailed data tables.

Calendar charts

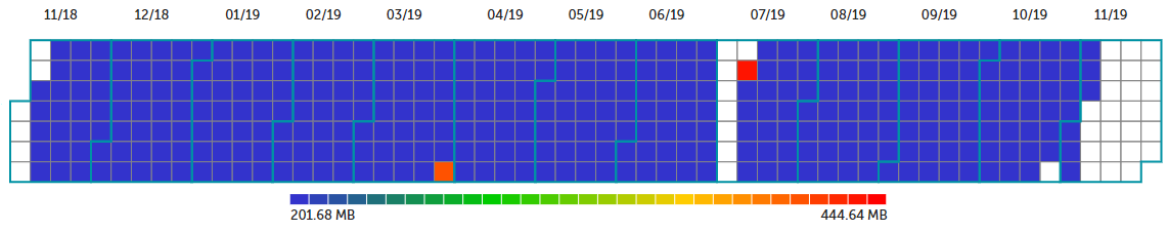


Figure 4.9: Calendar chart

This represents the real values of the events detected throughout a year.

Each box represents a day in each month. The boxes are grouped into blocks that represent the months of the year.

In turn, each box is colored according to the number of events in the day. The color range (blue - red) lets you quickly compare days against each other, thereby giving a better view of the development of the indicators monitored.

Move the mouse pointer over a box to see the corresponding color in the key, and a tooltip with the date and the exact number of events.

Stacked bar chart

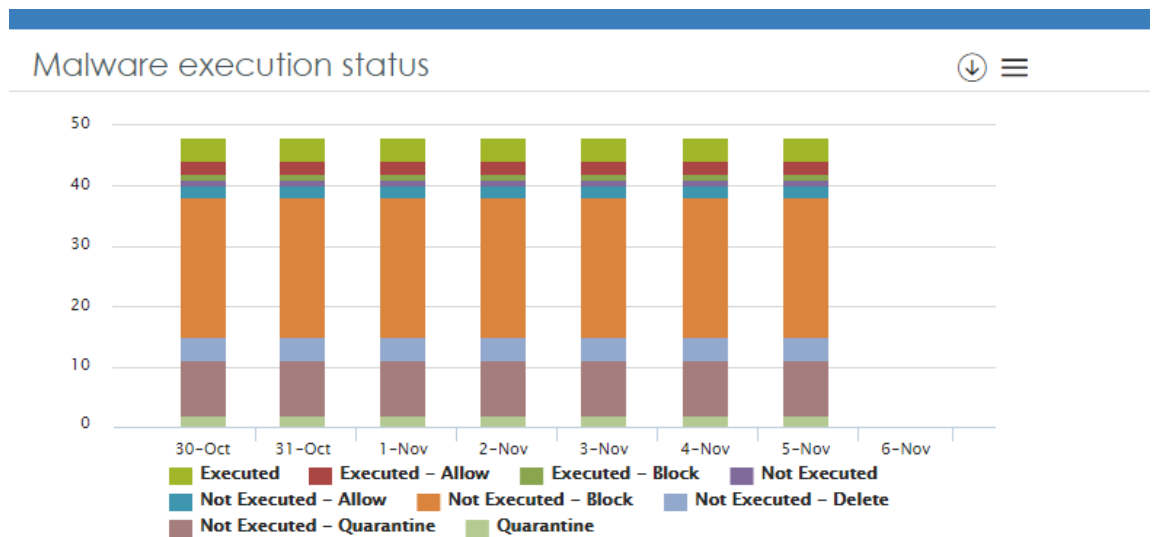


Figure 4.10: Stacked bar chart

Stacked bar charts show, in one chart, the evolution of multiple series, each represented with a color as indicated in the legend under the chart. They display, for each date, the contribution of each series to a total across categories.

Placing the mouse pointer over each series displays a tooltip indicating the date and time of the measurement and the value of the series at that time.

World map chart

This type of chart allows you to represent the values listed in the knowledge table on a map, provided the table contains 'Latitude' and 'Longitude' fields or data that can be used to provide coordinates.

The color and size of the points marked on the map (green-orange-red) indicate the relative number of events that have occurred in the established time period.

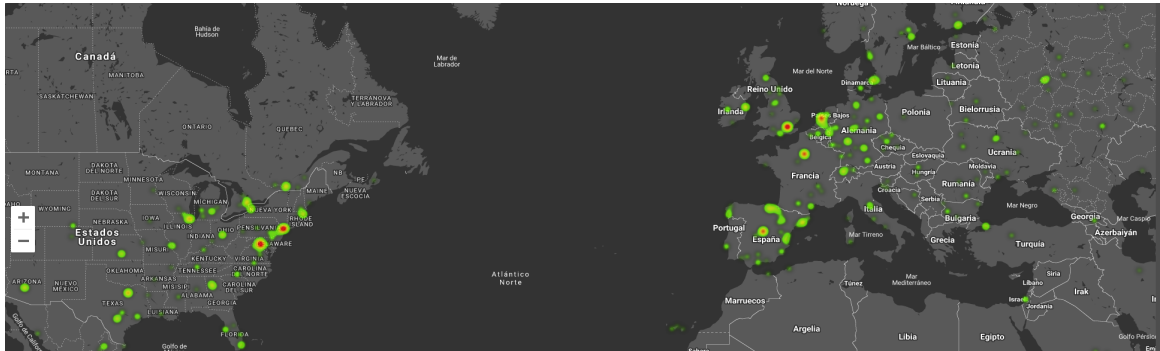


Figure 4.11: World Map chart

Voronoi diagram



Figure 4.12: Voronoi diagram and Thiessen polygons

A polygon can comprise other polygons representing groups of lower-level data.

As such there is a hierarchy of levels of groups ranging from the more general to the more specific. Voronoi diagrams allow you to navigate through the different levels of data groups: double-click using

the left mouse button on a group of data to access the lower level. From there, double-click using the right mouse button to return to the previous level.

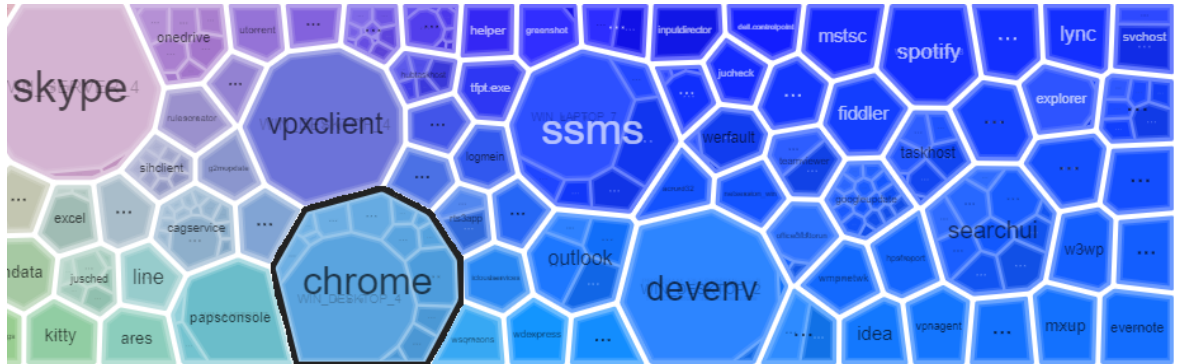


Figure 4.13: Zooming in into a polygon by double-clicking on it

Place the mouse pointer on a group to display the number of items in the group and the percentage that they represent of the total

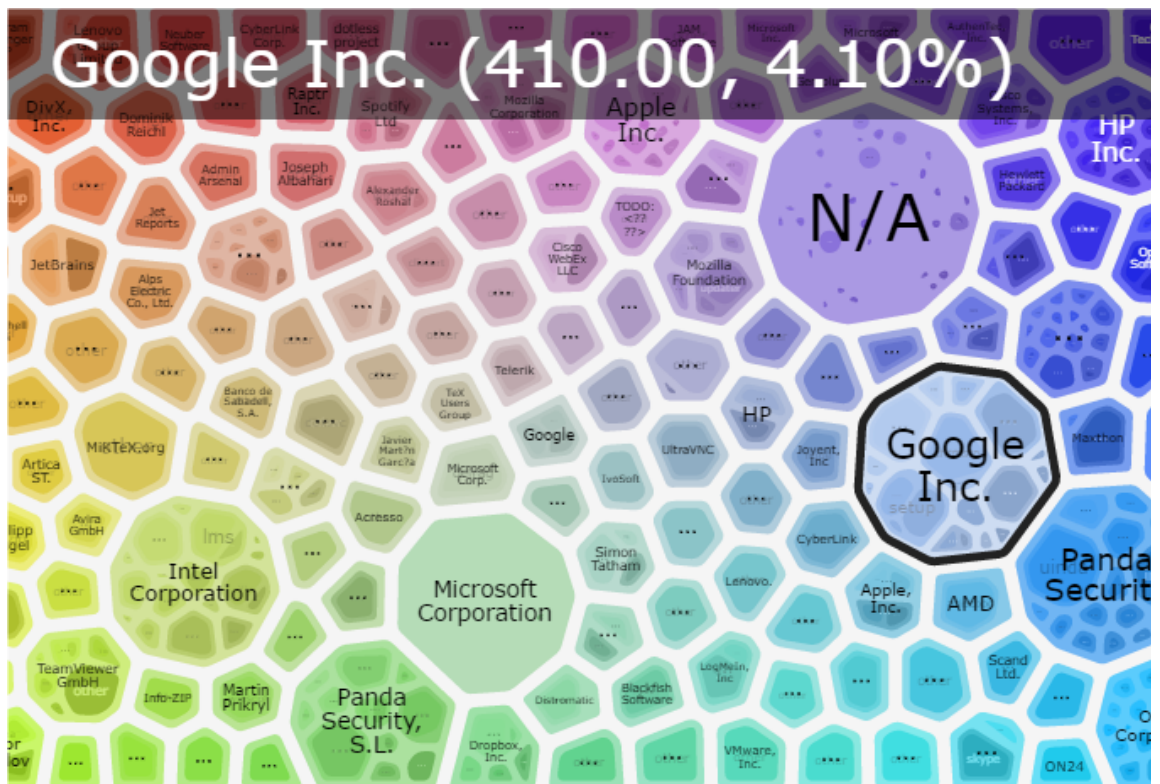


Figure 4.14: Data displayed within a polygon

A widget containing a Voronoi diagram offers the following controls:

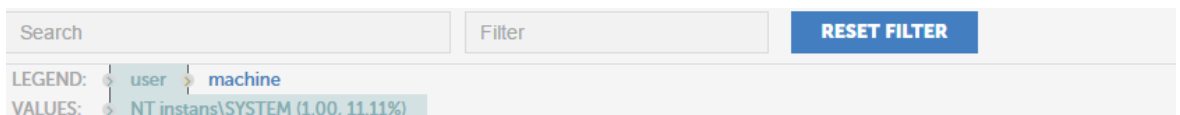


Figure 4.15: Controls for configuring the data displayed in a Voronoi diagram

- **Search:** This finds a polygon in the Voronoi diagram, and expands it to show the groups it comprises.

This is the same as double clicking with the left mouse button on a polygon in the diagram. To undo a search, double-click with the right mouse button.

- **Filter:** This shows the polygons that contain groups coinciding with the filter criteria.
- **Reset filter:** This clears the filter. It does not undo searches. To undo a search, double-click with the right mouse button.
- **Legend:** This indicates the knowledge table fields used to group the information displayed. The order of the fields indicates the group hierarchy and can be altered simply by dragging them to the left or right to establish a new hierarchy.
- **Values:** In combination with the fields shown in the **Legend** control, this indicates the value of a specific field. By selecting a polygon, either with the search tool, or by double-clicking it, the **Values** field will take the value of the search or the selected polygon.

Navigation by levels is carried out by double-clicking the left button on a Voronoi diagram polygon or by using the search tool. The highlighted field in **Legend** will take the value of the selected polygon, showing the next level of grouping indicated in the **Legend**.

- **Voronoi diagram example**

The following example illustrates how a Voronoi diagram works.

Depending on the **Legend**, the starting point is a chart that groups the data in the following order:

- **Level 1 AlertType:** Indicates the type of threat detected on the network.
- **Level 2 Machinename:** Indicates the name of the computer where the threat was detected.
- **Level 3 executionStatus:** Indicates whether or not it was executed.
- **Level 4 itemPath:** Indicates the file path and name.
- **Level 5 itemName:** Indicates the name of the threat.

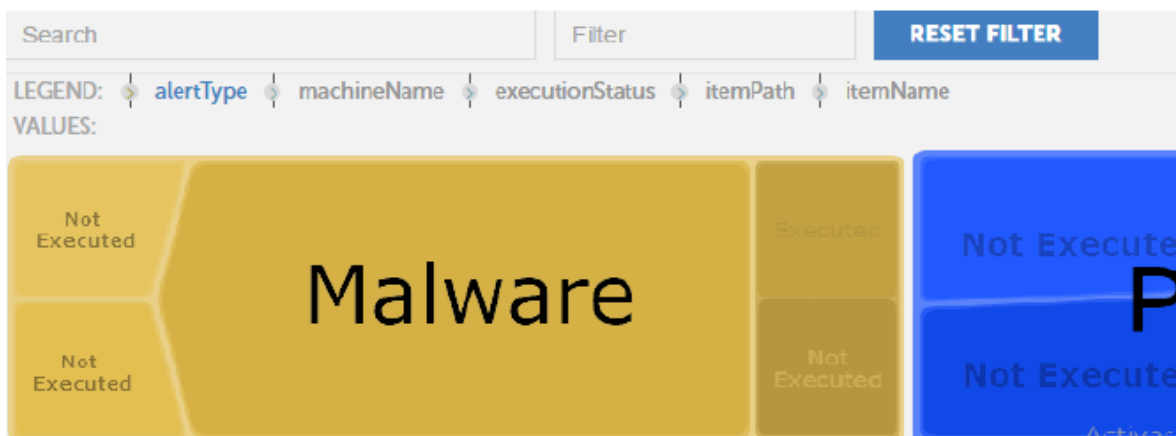


Figure 4.16: Example of the first data layer in a Voronoi diagram

At first, the diagram displays Level 1: the data grouped by **AlertType**, the first **Legend** field, highlighted in blue.

The second legend field is **MachineName**, so by double-clicking on one of the **AlertType** groups in the diagram (e.g. Malware) the second level will be displayed grouping the data according to **MachineName**. The Voronoi diagram will look like this:



Figure 4.17: Example of the second data layer in a Voronoi diagram

The **Values** field is refreshed displaying the **Level 1** selection (**AlertType=Malware**) and its content, the **Level 2**, with the data grouped by **MachineName**, highlighted in blue. Follow this process to navigate through the Voronoi diagram up to the last level, or move backwards through the diagram by double-clicking with the right mouse button.

If you want to establish an alternative order of grouping, simply drag the fields shown in **Legend** to the For example, if you want to first determine which computers have run some type of malware and then the name of the threat -in order to determine its characteristics-, then finally the computers on which it was executed, you can configure the grouping order as follows:

- Level 1 ExecutionStatus
- Level 2 ItemName

- Level 3 Machinename



Figure 4.18: New configuration for an alternative order of grouping

By double-clicking **Executed** in the Voronoi diagram, you can see the names of the items run; clicking one of these will display the computers on which it has been executed.

Pre-configured alerts

All the applications provided have pre-configured alerts that give administrators real-time information about any anomalous situations on the network.



See “Alerts” on page 55 for a description of the pre-configured alerts.

Accessing pre-configured alerts and setting the delivery frequency

The pre-configured alerts can be accessed through the side menu: **Administration, Alerts Configuration**.

Administrators have to complete the configuration of the alerts, setting the parameters below:

- **Alerts Subscriptions:** Go to the **Alerts Subscriptions** screen (**Administration, Alerts Configuration, Alerts Subscriptions** tab) to enable or disable the alerts. By default, all the pre-configured alerts are

enabled.

Alert Subscriptions
Subscribe to alerts that matter most to your system, reviews and modify its configuration

Delivery methods
Define where alerts are sent.

Alert Policies
Plan how and when you want to receive alerts as they occur.

Alerts Filter ⓘ

Adaptive Defense
Data control
My Alerts

alert
accesso
Accessed Data

FILTER **CLEAR FILTER**

Category	Subcategory	Alert	Owner	Active Policies
Adaptive Defense	Security Incident	Malware per endpoint hourly	-	Konsult dagtid ON
Adaptive Defense	Data Access Control	Users and Outbound data hourly	-	default OFF

Figure 4.19: Alerts Subscriptions

- **Alert receipt frequency:** Administrators have to configure post filters (**Alerts, Post filters** tab) and anti-flooding policies (side menu **Administration, Alerts Configuration, tab Alert Policies, Anti-flooding policies** tab) explained in Chapter 6 to set the frequency with which alerts are generated to the administrator's needs.
- **Delivery methods:** Administrators have to configure the methods used to deliver the alerts (Email, Json or others) in accordance with the company's infrastructure, explained in "**Alerts**" on page 55. You can access these settings by clicking **Administration, Alerts Configuration, Delivery methods** tab.



There is no limit to the amount of alerts generated by Panda Advanced Reporting Tool. The alerts will only be displayed in the Web console in the Alerts section of the side menu, until the configuration described above has been carried out.

Generating new charts based on the widgets provided

By clicking the ☰ icon in each widget and selecting **Go to Search**, the corresponding knowledge table that feeds that widget will open.

Each knowledge table has a series of transformations, filters and groups designed to present the most important data clearly and accurately. These transformations are in SQL language and can be edited to adapt to the customer's needs.



It is not possible to overwrite the widgets provided, but you can generate new widgets using the original ones as a base.

Modifying the SQL statement associated with a widget

Once you are in the knowledge table associated with a widget, click the 🗄 icon in the toolbar. A window with the preset SQL statement will open. After editing the statement, click **Run** to test the execution. The data in the table will be updated immediately.

You can also modify the SQL statement by adding new filters, groups and data transformations via the toolbar.

SQL statement favorites

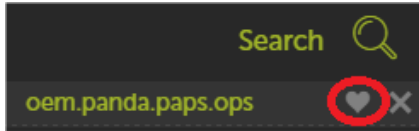


Figure 4.20: Mark an alert as a favorite

After changing the SQL statement and ensuring that the generated data is correct, it can then be saved for later access, by marking it as a **Favorite**. Opening a knowledge table will display a new entry in the sidebar, below the **Search** icon. A heart icon will be displayed to the right of the name of the entry.

Click this icon and the SQL statement will be marked as **Favorite**, and will appear in the list of favorites

Favorites can be found in the side menu **Administration, Alerts Configuration**.

Chapter 5

Configured applications

This chapter describes how the three applications provided with Panda Advanced Reporting Tool operate, both regarding the interpretation of charts and tables as well as the operation of the pre-configured alerts.

CHAPTER CONTENT

Setting the time period	-36
Wider ranges of dates	36
Narrower date ranges	37
Associated alerts	-37
Security Incidents application	-38
Key Security Indicators	38
Alerts summary (daily or weekly)	38
Malware-PUP-Exploit execution status	38
Calendar of Daily Malware Detections	39
Calendar of Daily Potential Unwanted Programs (PUPS) Detections	39
Calendar of daily exploit detections	39
Detailed Information	39
Endpoints involved in Incidents	40
Incidents in All Endpoints	40
Malware per endpoint hourly	40
Malware in the network hourly	40
Malware executed in different endpoints hourly	41
Application Control application	-41
IT Applications	41
Executed Applications	41
Most frequently executed applications	42
Machines running the most frequently executed applications	42
Least frequently executed applications	42
Machines running the least frequently executed applications	42
Microsoft Office Licenses in use	43
Microsoft Office Applications in use	43
Microsoft Office Applications by user	43
Vulnerable Applications	43
Vulnerable applications installed	44
Vulnerable applications installed by machine	44
Vulnerable applications executed	44
Vulnerable applications executed by machine	44
Bandwidth-consuming Applications	45
Data Volume Received by applications	45
Data Volume Sent by applications	45
Special Applications & Tools	46
Scripting Applications Executed	46

- Scripting Applications Executed by machine and user46
- Remote Access Applications Executed46
- Remote Access Applications Executed by machine and user47
- Admin Tools Executed47
- Admin Tools Executed by machine and user47
- System Tools Executed47
- System Tools Executed by machine and user48
- System Internal Tools Executed48
- System Internal Tools Executed by machine and user48
- Unwanted Freeware Applications Executed48
- Unwanted Freeware Applications Executed by machine and user49
- Executions of Vulnerable apps per endpoint today49
- Bandwidth consumption to endpoint hourly49
- Bandwidth consumption from endpoint hourly49
- Bandwidth consumption per apps hourly50
- Data Access Control Application - - - - - 50**
- Outbound network traffic50
 - Annual Calendar of outbound network traffic50
 - Countries with outbound connections51
 - Outbound network traffic destinations51
- User activity51
 - Logged-in users51
- Bandwidth consumers52
 - Applications with most inbound network traffic52
 - Applications with most outbound network traffic52
 - Machine-User pairs with most inbound network traffic52
 - Machine-User pairs with most outbound network traffic52
- Data Files Accessed53
 - Files most accessed from endpoints53
 - Most accessed files by user53
 - Most executed extensions53
 - Users and Outbound data hourly53

Setting the time period

The three applications provided have a control option at the top of the screen to allow you to set the data time period.



Figure 5.1: Date range picker

Administrators have to set the date range to view the security status of the network.

Wider ranges of dates

When the date range set is wider (months or days), the data will be displayed as a history or an evolution of activity over time.

- **Execution of unknown threats and vulnerable applications**

If the network administrator has configured an advanced protection mode in Panda Adaptive Defense other than **Lock** (i.e. **Audit** or **Hardening**), it is possible for a user to run unknown malware. This

threat would continue to run on the user's computer until the issue is resolved. For this reason, the execution of an unknown threat is an event that continues over time. If the date range selected in Panda Advanced Reporting Tool covers the period of execution of the threat, it will be shown in the charts as executed malware, even if the situation has already been successfully resolved.

- **Blocking of known threats**

Where there is an attempt to run known malware (blocking), detections occur at a specific point in time. If the date range selected by the administrator includes this point in time, the detection is displayed.

Narrower date ranges

By selecting a narrower range of dates, such as the current day, administrators can determine the current status of network security but will lose the perspective of data over time.

- **Execution of unknown threats and vulnerable applications**

If unknown malware was executed in the past and has not yet been resolved, the malware will be displayed in the graphs as executing. This means that administrators can quickly determine if there are any issues pending resolution.

- **Blocking of known threats**

By selecting date ranges as the current day, only infection attempts by known threats on that day will be displayed.

Associated alerts

To access the alerts associated with the applications, click **Administration** from the side menu and then click **Alert Configuration**.

To configure an associated alert, select **Adaptive Defense** from the panel on the left and the relevant application from the panel on the right (**Application Control**, **Data Access Control** or **Security Incidents**). The bottom list will display the different types of available alerts:

- The **Security Incidents** application alerts inform the administrator of malware detection events.
- The **Application Control** application alerts inform the administrator of the execution of vulnerable applications and bandwidth consumption, as part of the IT Department's proactive strategy to ensure proper operation of the network.
- The **Data Access Control** alerts inform the administrator of the amounts of data sent by the users of the managed network.

Security Incidents application

Security Incidents lets you see malware activity on customers' networks and adapt the organization's security policies accordingly. It can also help generate baseline information for forensic analysis.

The dashboard shows detections across the network and related information:

- **Information about computers affected:** Number of detections, evolution of detections over time, etc.
- **Information about threats detected:** Infection vectors, computers affected, execution status of the virus, type of virus, etc.

The dashboard is divided into two tabs: **Key Security Indicators** and **Detailed Information**. These are explained below.

Key Security Indicators

This tab gives an overview of the most important data about malware activity on the network.

It is divided into two sections:

- **Incidents:** This shows data about the type of malware detected, the computers affected, whether or not the threat was executed and other relevant information.
- **Malware, PUPS and Exploits:** Show the evolution of detections on the network. This information is displayed through calendar-type widgets.

Alerts summary (daily or weekly)

- **Malware and PUPS:** Show the incidents detected in the processes run on users' workstations and in their file systems. These incidents are reported both by the real-time scans and on-demand scans.
- **Exploits:** Shows the number of vulnerability exploit attacks suffered by the Windows computers on the network.

The alerts use arrows and percentages to show the variation in the number of detected incidents compared to the previous day (daily) and previous week (weekly).

Malware-PUP-Exploit execution status

- **Aim:** To display the evolution of malware detected on the customer's network.
- **Type of widget:** Stacked bar chart.
- **Data displayed:** Number of malware detections on all network computers, grouped by day of the month.
- **Grouping:** Day of the month.

This widget uses color codes to rapidly depict the days of the year on which most malware detections have occurred on the customer's network. In this way, it allows you to identify 'black days'

and investigate the causes.

Calendar of Daily Malware Detections

- **Aim:** To display the evolution of detections of malware on the customer's network.
- **Type of widget:** Calendar chart.
- **Data displayed:** Number of detections of malware on all network computers, grouped by day of the month.
- **Grouping:** Day of the month.

This widget uses color codes to rapidly depict the days of the year on which most detections of malware have occurred on the customer's network. In this way, it allows you to identify 'black days' and investigate the causes.

Calendar of Daily Potential Unwanted Programs (PUPS) Detections

- **Aim:** To display the evolution of detections of **Potential Unwanted Programs (PUP)** on the customer's network.
- **Type of widget:** Calendar chart.
- **Data displayed:** Number of detections of **Potential Unwanted Programs (PUP)** on all network computers, grouped by day of the month.
- **Grouping:** Day of the month.

This widget uses color codes to rapidly depict the days of the year on which most detections of Potential Unwanted Programs (PUP) have occurred on the customer's network. In this way, it allows you to identify 'black days' and investigate the causes.

Calendar of daily exploit detections

- **Aim:** To display the evolution of the exploit-type threats found on the customer's network.
- **Type of widget:** Calendar chart.
- **Data displayed:** Number of exploit detections on all computers on the network, grouped by day of the month.
- **Grouping:** Day of the month.

This widget uses color codes to rapidly show the days of the year on which most exploit detections have occurred on the customer's network. This way, it allows you to identify 'black days' and investigate the causes.

Detailed Information

This contains an **Incidents** section which uses several tables to indicate the incidents caused by

malware.

Endpoints involved in Incidents

- **Aim:** To help locate the network computers with most threats detected, and their type.
- **Fields:**
 - **Alert type:** Type of threat (Malware or PUP).
 - **Machine name:** Name of the computer on which the threat was detected.
 - **Alert count:** Counter with the number of incidents over a set period.

This table can help rapidly locate computers that may have a higher probability of causing network problems.

Incidents in All Endpoints

- **Aim:** To show a complete list of all endpoints infected over the selected period, including all relevant information.
- **Fields:**
 - **Alert type:** Type of threat (Malware, PUP, Exploit).
 - **Machine name:** Name of the computer on which the threat was detected.
 - **Execution status:** Indicates whether the threat was run or not (Executed | not Executed).
 - **Program:** Full path to the detected threat.
 - **Threat:** Name of the threat.
 - **Threat count:** Counter with the number of incidents over the time period.

Malware per endpoint hourly

- **Aim:** To show the number of malware detections in the last hour on each network computer.
- **SQL:**

```
from oem.panda.paps.alert where alertType = "Malware" group every 30m by  
machineName every 0 select count() as count
```

Malware in the network hourly

- **Aim:** To show the number of malware detections in the last hour on the whole network.
- **SQL:**

```
from oem.panda.paps.alert where alertType = "Malware" group every 30m every 0  
select count() as count
```

Malware executed in different endpoints hourly

- **Aim:** To show the number of computers that have executed a certain type of malware in the last hour.

SQL:

```
from oem.panda.paps.alert where alertType = "Malware", executionStatus = "Executed" group every 30m every 0 select count() as count
```



See "[Associated alerts](#)" for more information.

Application Control application

Application Control offers detailed information about the applications installed and run on users' computers.

The dashboard is divided into four tabs: IT Applications, Vulnerable Applications, Bandwidth-consuming Applications and **Special Applications & Tools**.

IT Applications

This tab allows administrators to find out which applications ran on network computers, as well as establish basic control over the Microsoft Office licenses in use.

Executed Applications

- **Aim:** To show as a percentage the software developers whose applications are running on the network, the name of the executable, the path where it is located on the hard drive of the user's computer, and the computer on the network that ran it.
- **Type of widget:** Voronoi diagram.
- **Data displayed:**
 - **First level:** Name of the developer of the executed software.
 - **Second level:** Name of the program.
 - **Third level:** Full path of the executed program on the hard disk of the user's computer.
 - **Fourth level:** Name of the computer that executed the program.
 - **Grouping:** Company name, software name, path, computer.

This chart allows administrators to quickly identify the most frequently executed programs on the network, in order to detect the use of inappropriate or unlicensed software.

Most frequently executed applications

- **Aim:** To show a list of those applications most frequently run across the network.
- **Fields:**
 - **Childpath:** Full path to the executed application.
 - **Executable:** Executable file name.
 - **Count:** Number of occurrences in the selected time range.
 - **%:** Executions of the program as a percentage of the total number of program executions across the network.

Machines running the most frequently executed applications

- **Aim:** To show a list of those computers running the most frequently executed applications.
- **Fields:**
 - **Childpath:** Name of the computer where the application was run.
 - **Executable:** Executable file name.
 - **Count:** Number of times the application was run on the computer in the selected time range.
 - **%:** Executions of the program as a percentage of the total number of program executions across the network.

Least frequently executed applications

- **Aim:** To show a list of those applications least frequently run across the network. It provides visibility into executed applications that may fall under the radar of the IT department.
- **Fields:**
 - **Childpath:** Full path to the executed application.
 - **Executable:** Executable file name.
 - **Count:** Number of times the application was run in the selected time range.

%. Executions of the program as a percentage of the total number of program executions across the network.

Machines running the least frequently executed applications

- **Aim:** To show a list of those computers running the least frequently executed applications.
- **Fields:**
 - **Machine:** Name of the computer where the application was run.
 - **Executable:** Executable file name.
 - **User Count:** Number of users who ran the application on the computer in the selected time range.

- **Execution Count:** Number of times the application was run on the computer in the selected time range.
- **%:** Executions of the program as a percentage of the total number of program executions across the network.

Microsoft Office Licenses in use

- **Aim:** To show the Microsoft Office applications used across the network and the number of users who ran them.
- **Type of widget:** Voronoi diagram.
- **Fields:**
 - **First level:** Name of the Microsoft Office application run.
 - **Second level:** Number of users who ran the application in the selected time range.
 - **Grouping:** Name of the Microsoft Office application, user.

Microsoft Office Applications in use

- **Aim:** To show the Microsoft Office applications used across the network and the number of users who ran them.
- **Fields:**
 - **Office Application:** Name of the Microsoft Office application run.
 - **Count:** Number of users who ran the application in the selected time range.
 - **%:** Executions of the program as a percentage of the total number of program executions across the network.

Microsoft Office Applications by user

- **Aim:** To show the number of users who ran an application belonging to the Microsoft Office suite and the number of executions.
- **Fields:**
 - **User:** Name of the user who ran the Microsoft Office application.
 - **Office Applications in Use Count:** Number of times the user ran a Microsoft Office application.
 - **%:** Executions of the program as a percentage of the total number of program executions across the network.

Vulnerable Applications

This tab allows administrators to determine the vulnerable applications installed and/or executed on network computers. The purpose of the charts is to establish the IT department's priorities when updating software with known vulnerabilities.

Vulnerable applications installed

- **Aim:** To show the number of vulnerable software applications installed across the network.
- **Fields:**
 - **Vulnerable application:** Name of the vulnerable software application.
 - **Machine count:** Number of computers where the vulnerable application was installed/run.
 - **%:** Installations/executions of the program as a percentage of the total number of program installations/executions across the network.

Vulnerable applications installed by machine

- **Aim:** To show the vulnerable software applications installed across the network and the computers where they are installed.
- **Fields:**
 - **Vulnerable application:** Name of the vulnerable software application.
 - **Company:** Vendor of the vulnerable software application.
 - **Machine:** Name of the computer where the vulnerable application is installed.
 - **%:** Installations/executions of the program on the computer as a percentage of the total number of program installations/executions across the network.

Vulnerable applications executed

- **Aim:** To show the vulnerable software applications run across the network.
- **Fields:**
 - **Vulnerable application:** Name of the vulnerable software application.
 - **Machine count:** Number of computers where the vulnerable application was run.
 - **%:** Executions of the program as a percentage of the total number of program executions across the network.

Vulnerable applications executed by machine

- **Aim:** To show the vulnerable software applications run across the network, their vendor, and the computer on which each application was run.
- **Fields:**
 - **Company:** Vendor of the software application run.
 - **Machine:** Name of the computer where the vulnerable application was run.
 - **Count:** Number of times the vulnerable application was run on the computer.
 - **%:** Executions of the program on the computer as a percentage of the total number of program executions across the network.

Bandwidth-consuming Applications

This tab displays the volume and percentage of bandwidth consumed by the applications running on the network. The aim is to provide an overview of the bandwidth consumption of the applications executed by users with two aims: to detect applications with above average consumption, and to help ensure optimum dimensioning of bandwidth provisioning across the organization.

Data Volume Received by applications

This shows the volume and the percentage of bandwidth received by each application running on the network, along with the path of the application and the computer on which it was run.

- **Aim:** To show the volume and the percentage of bandwidth received by each application running on the network, the path of the application and the computer on which it was run.
- **Type of widget:** Voronoi diagram.
- **Data displayed:**
 - **First level:** Executable that receives the data.
 - **Second level:** Name of the computer receiving the data.
 - **Third level:** Full path of the executable on the customer's computer.
 - **Grouping:** Executable, computer name, path.

An alternative grouping that would help to view the computers that receive most traffic on the network would be: machineName, executable, path.

Data Volume Sent by applications

This shows the volume and the percentage of bandwidth sent by each application running on the network, along with the path of the application and the computer on which it was run.

- **Aim:** To show the volume and the percentage of bandwidth sent by each application running on the network, the path of the application and the computer on which it was run.
- **Type of widget:** Voronoi diagram.
- **Data displayed:**
 - **First level:** Executable that sends the data.
 - **Second level:** Name of the computer receiving the data.
 - **Third level:** Full path of the executable on the customer's computer.
 - **Grouping:** Executable, computer name, path.

An alternative grouping that would help to view the computers that send most traffic on the network would be: machineName, executable, path.

Special Applications & Tools

This tab details the scripting, remote access, administrator and system tools as well as unwanted free software applications running on the network. In addition, it is specified which of these applications and tools have been run by equipment / user and the number of times this has happened.



Refer to <https://www.pandasecurity.com/support/card?id=700065> for a list of programs detected by ART.

Scripting Applications Executed

- **Aim:** To show the script interpreters or engines run across the network.
- **Fields:**
 - **Scripting Application:** Name of the scripting application run.
 - **Machine count:** Number of computers where the scripting application was run.
 - **%:** Executions of the application as a percentage of the total number of application executions across the network.

Scripting Applications Executed by machine and user

- **Aim:** To show the scripting applications run by users on each computer.
- **Fields:**
 - **Machine:** Name of the computer where the scripting application was run.
 - **User:** Name of the user who ran the scripting application on the computer.
 - **Scripting application:** Name of the scripting application run.
 - **Scripting application count:** Number of times the scripting application was run on the computer.
 - **%:** Executions of the application on the computer as a percentage of the total number of application executions across the network.

Remote Access Applications Executed

- **Aim:** To show the remote access applications run across the network.
- **Fields:**
 - **Remote Access Application:** Name of the remote access application run.
 - **Machine count:** Number of computers where the remote access application was run.
 - **%:** Executions of the remote access application as a percentage of the total number of application executions across the network.

Remote Access Applications Executed by machine and user

- **Aim:** To show the remote access applications run by users on each computer.
- **Fields:**
 - **Machine:** Name of the computer where the remote access application was run.
 - **User:** Name of the user who ran the remote access application.
 - **Remote Access application:** Name of the remote access application run.
 - **Remote application count:** Number of times the remote access application was run by the user on the computer.
 - **%:** Executions of the remote access application on the computer as a percentage of the total number of application executions across the network.

Admin Tools Executed

- **Aim:** To show the admin tools run across the network.
- **Fields:**
 - **Admin Tools Executed:** Name of the admin tool run.
 - **Machine count:** Number of computers where the admin tool was run.
 - **%:** Executions of the admin tool as a percentage of the total number of executions recorded across the network.

Admin Tools Executed by machine and user

- **Aim:** To show the admin tools run by users on each computer.
- **Fields:**
 - **Machine:** Name of the computer where the admin tool was run.
 - **User:** Name of the user who ran the admin tool.
 - **Admin Tools Executed:** Name of the admin tool run.
 - **Admin Tool Count:** Number of times the admin tool was run by the user on the computer.
 - **%:** Executions of the admin tool on the computer as a percentage of the total number of executions recorded across the network.

System Tools Executed

- **Aim:** To show the system tools run across the network.
- **Fields:**
 - **System Tools Executed:** Name of the system tool run.
 - **Machine count:** Number of computers where the system tool was run.

- %: Executions of the system tool as a percentage of the total number of executions recorded across the network.

System Tools Executed by machine and user

- **Aim:** To show the system tools run by users on each computer.
- **Fields:**
 - **Machine:** Name of the computer where the system tool was run.
 - **User:** Name of the user who ran the system tool.
 - **System Tools Executed:** Name of the system tool run.
 - **System Tool Count:** Number of times the system tool was run by the user on the computer.
 - %: Executions of the system tool on the computer as a percentage of the total number of executions recorded across the network.

System Internal Tools Executed

- **Aim:** To show the system internal tools run across the network.
- **Fields:**
 - **System Internal Tools:** Name of the system internal tool run.
 - **Machine count:** Number of computers where the system internal tool was run.
 - %: Executions of the system internal tool as a percentage of the total number of executions recorded across the network.

System Internal Tools Executed by machine and user

- **Aim:** To show the system internal tools run by users on each computer.
- **Fields:**
 - **Machine:** Name of the computer where the system internal tool was run.
 - **User:** Name of the user who ran the system internal tool.
 - **System Internal Tools:** Name of the system internal tool run.
 - **System Internal Tool Count:** Number of times the system internal tool was run by the user on the computer.
 - %: Executions of the system internal tool on the computer as a percentage of the total number of executions recorded across the network.

Unwanted Freeware Applications Executed

- **Aim:** To show the unwanted freeware applications run across the network.
- **Fields:**

- **Freeware Application:** Name of the unwanted freeware application run.
- **Machine count:** Number of computers where the unwanted freeware application was run.
- **%:** Executions of the unwanted freeware application as a percentage of the total number of executions recorded across the network.

Unwanted Freeware Applications Executed by machine and user

- **Aim:** To show the unwanted freeware applications run by users on each computer.
- **Fields:**
 - **Freeware Application:** Name of the unwanted freeware application run.
 - **Machine:** Name of the computer where the unwanted freeware application was run.
 - **User:** Name of the user who ran the unwanted freeware application on the computer.
 - **Freeware application count:** Number of times the unwanted freeware application was run by the user on the computer.
 - **%:** Executions of the unwanted freeware application on the computer as a percentage of the total number of executions recorded across the network.

Executions of Vulnerable apps per endpoint today

- **Aim:** To show the number of vulnerable applications run in the last 24 hours by each network computer.
- **SQL:**

```
from oem.panda.paps.ops where isnotnull(ocsVer) group every 30m by machine every 1d
select count(childPath) as childPath
```

Bandwidth consumption to endpoint hourly

- **Aim:** To show the bandwidth received in the last hour by each network computer.
- **SQL:**

```
from oem.panda.paps.processnetbytes group every 30m by machineName every 0
select sum(bytesReceived) as sum_bytes_received
```

Bandwidth consumption from endpoint hourly

- **Aim:** To show the bandwidth sent in the last hour by each network computer.
- **SQL:**

```
from oem.panda.paps.processnetbytes group every 30m by machineName every 0
select sum(bytesSent) as sum_bytes_sent
```

Bandwidth consumption per apps hourly

- **Aim:** To show the bandwidth received and sent in the last hour by each app.
- **SQL:**

```
from oem.panda.paps.processnetbytes select subs(path,
re("(.*\\\\\\\\) (?=.*(\\\\.\\\\w*)$|\\\\w+$)"), template("")) as executablename select
lower(executablename) as executable where endswith(executable, "exe") group
every 15s by executable every 15s select sum(bytesReceived) as
sum_bytes_consumption
```



See "Malware per endpoint hourly" for more information.

Data Access Control Application

Data Access Control displays the information that leaves the customer's network in order to detect data leaks and theft of confidential information.

The dashboard is divided into four tabs: **Outbound network traffic**, **Users activity**, **Bandwidth consumers** and **Data file accessed**.

Outbound network traffic

This tab displays information about the volume of data sent out from the customer's network. It is divided into two sections:

- **Data:** This shows absolute and relative values of the transfer of data.
- **Map:** This displays geolocation on a world map of the destinations to which the greatest percentage of data has been sent.

Annual Calendar of outbound network traffic

- **Aim:** This shows the evolution of data sent from the customer's network.
- **Type of widget:** Calendar chart.
- **Data displayed:** The volume of data sent -in megabytes or gigabytes- from all computers on the customer's network, grouped by day of the month.
- **Grouping:** Day of the month.

This graph lets administrators locate the days of the month during which network computers have sent an abnormally high volume of data.

Countries with outbound connections

- **Aim:** To identify the ten countries that have received most connections from the customer's network.
- **Fields:**
 - **CC:** Country code of the target country.
 - **Count:** Number of connections.
 - **%:** Volume of connections to each country as a percentage.

This chart identifies the 10 countries that have received most connections from the network. In these cases, a strong indication of potential problems is when there are countries on the list with which the company does not normally have a commercial relation.

Outbound network traffic destinations

- **Aim:** To geolocate on a map the destinations of the organization's network traffic.
- **Type of widget:** Map chart.
- **Data displayed:** A representation of the volume of data sent from the customer's network to the countries indicated in the map by dots of different intensity. The color and diameter of the dots represent the volume of data sent.
- **Grouping:** Country.

In addition to the **Countries with outbound connections** table, there is a map with the countries that have received data from the customer's network, showing the relative volume of traffic.

User activity

This displays information about network user activity.

Logged-in users

- **Aim:** To show the computers accessed by each user account on the network.
- **Type of widget:** Voronoi diagram.
- **Data displayed:**
 - **First level:** User accounts.
 - **Second level:** Computers accessed by the user accounts selected in the first level.
 - **Grouping:** User, computer.

A possible variation to this graph can be obtained by changing the order of the **Legend** field to **machine, user**, if you want to determine which user accounts have accessed each computer.

Bandwidth consumers

This identifies the processes and users that have consumed most network bandwidth. Tables contain a maximum of 1,000 records, sorted in descending order.

Applications with most inbound network traffic

- **Aim:** To show those applications that receive most inbound traffic on the customer's network.
- **Fields:**
 - **Executable:** Name of the executable file that receives the data.
 - **Volume received:** The sum of the volume of data received.
 - **%:** Volume of data received as a percentage of the total.

Applications with most outbound network traffic

- **Aim:** To show those applications that send most outbound traffic on the customer's network.
- **Fields:**
 - **Executable:** Name of the executable file that sent the data.
 - **Volume sent:** The sum of the volume of data sent.
 - **%:** Volume of data sent as a percentage of the total.

Machine-User pairs with most inbound network traffic

- **Aim:** To show the machine-user pairs that receive most inbound traffic on the customer's network.
- **Fields:**
 - **User:** User logged in to the computer that sends the traffic.
 - **Machinename:** Name of the computer that sends the traffic.
 - **Volume received:** Volume of data sent.
 - **%:** Volume of data sent as a percentage of the total.

Machine-User pairs with most outbound network traffic

- **Aim:** To show the machine-user pairs that send most outbound traffic on the customer's network.
- **Fields:**
 - **User:** User logged in to the computer that receives the traffic.
 - **Machinename:** Name of the computer that receives the traffic.
 - **Sum_sent_sum:** Volume of data received.
 - **%:** Volume of data received as a percentage of the total.

Data Files Accessed

This identifies the files accessed by users of the customer's network. With the data provided in this section, administrators have access to some DLD (Data Leak Detection) features. Tables contain a maximum of 1,000 records, sorted in descending order.

The following sections are available:

- **Endpoints:** This displays file access statistics by user and extension
- **Users & Extensions:** This displays file access statistics by file extension

Files most accessed from endpoints

- **Aim:** To display the files most accessed by network users.
- **Fields:**
 - **Machine:** Name of the computer used to access the file.
 - **Childpath:** Path and name of the file.
 - **Count:** Number of times the computer has accessed the file.
 - **%:** Access to the file as a percentage of the total number of file accesses.

Most accessed files by user

- **Aim:** To display the files most accessed by network users.
- **Fields:**
 - **Childpath:** Path and name of the file.
 - **Logged-in user:** The logged-in user accessing the file.
 - **Count:** Number of times the user has accessed the file.
 - **%:** Access to the file as a percentage of the total number of file accesses.

Most executed extensions

- **Aim:** To display the extensions most frequently run on the network, either individually (executable file extensions), or as data files opened by programs (Office files, compressed files, etc.)
- **Fields:**
 - **File Extension:** File extension.
 - **Count:** Number of times a file with that extension has been accessed.
 - **%:** Volume of accesses as a percentage of the total.

Users and Outbound data hourly

- **Aim:** To display the volume of data sent by each user in the last 24 hours.

- **SQL:**

```
from oem.panda.paps.processnetbytes select yesterday("") as yest_date where
eventdate >= yest_date and not startswith(user, "NTAUTHORITY") and not
startswith(user, "<unknown>\\<unknown>") groupevery 30m by user every 1d select
sum(bytesSent) as total_tx
```



See "[Associated alerts](#)" for more information.

Chapter 6

Alerts

The Panda Advanced Reporting Tool alerts system allows administrators to keep up-to-speed with events that take place on the network that require their attention, without having to go to the Web console. It is therefore a key module in minimizing the reaction time of the IT department when faced with potentially dangerous situations for the organization.

The alerts system is fully configurable by the network administrator, including the frequency for sending alerts, the conditions required for generating them and the delivery method used.

CHAPTER CONTENT

Alert system architecture	-56
Process for configuring the alerts	56
Creating alerts	-57
Alert management	59
Alerts Overview	59
Alerts History	60
Establishing filters in the alerts history	60
Creating post filters	-61
Section 1: Description	61
Section 2: Basic data	61
Section 3: Extra data	61
Section 4: Filter dates	62
Section 5: Action	62
Post filter management	62
Creating delivery conditions	-63
Email	63
HTTP-JSON	63
Service desk	64
JIRA	64
Pushover	65
Pagerduty	66
SLACK	66
Delivery method management	66
Creating antiflooding policies	-67
Editing antiflooding policies	67
Creating alert policies or delivery methods	-67
Editing sending policies	68
Configuring an alert sending policy	68

Alert system architecture

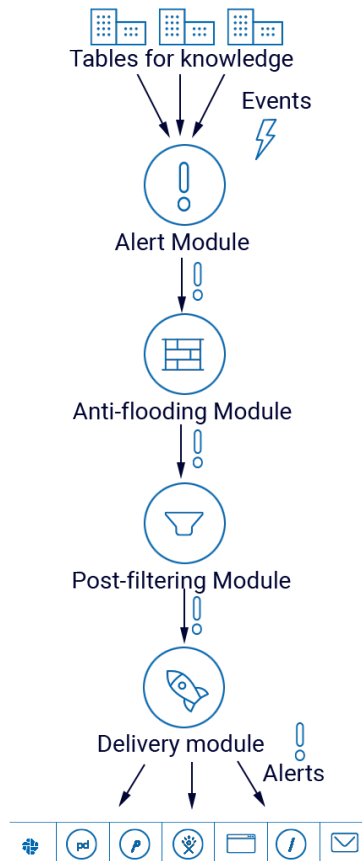


Figure 6.1: Modules implemented in the alert generation flow

The Panda Advanced Reporting Tool alerts system comprises several fully configurable modules. The sequence of processes involved in the generation of alerts is as follows:

- **Generation of events:** Each entry in a **knowledge table** generates a unique event that can later be converted into one or more alerts.
- **Alerts module:** The events that meet certain criteria defined by administrators in the alerts module will generate an alert.
- **Antiflooding module:** This prevents the problem of a 'storm of alerts', allowing the alerts generation module to be temporarily disconnected from the generation of events on exceeding a certain threshold defined by the administrator. This prevents the generation of a flood of alerts.
- **Post filter module:** This handles the alerts once they are generated, changing their properties or even selectively eliminating them in line with the criteria established by the administrator.
- **Delivery module:** This allows the delivery of the alerts to administrators in a number of ways: email, HTTP-JSON, Service Desk, Jira, Pushover, PagerDuty, and Slack. For more information, refer to section "[Creating delivery conditions](#)".

Process for configuring the alerts

Setting up a new alert requires a series of steps, some of them mandatory, some of them optional, in order for the alert to work correctly.

These steps are listed below along with a brief description of the process.

1. **Creating the alerts (mandatory):** Creating an alert requires you to define the type of event you want from the knowledge table, and to establish that it will generate an alert.
2. **Editing the alert subscription (optional):** This lets you enable or disable the newly created alert. Alerts are enabled automatically when they are created.
3. **Set the delivery criteria (mandatory for the first alert):** The delivery settings allow you to determine the delivery method and specify associated information. For example, if you specify delivery by email, you must indicate the recipient's email account.
4. **Creating an antiflooding policy (optional):** This sets maximum thresholds for generating alerts in order to avoid mass mailings. Administrators who prefer to receive all generated alerts shouldn't use

any antiflooding policy.

5. **Creating a new delivery policy (mandatory for the first alert):** The delivery policy lets you define the following parameters for delivering alerts:
 - **Assigning the antiflooding policy** (point 4).
 - **Assigning the delivery schedule:** Alerts will only be sent in line with the calendar settings.
 - **Delivery method** (point 3).
6. **Assigning a delivery policy** (point 5) to the alert created (point 1).
7. **Creating post filters (optional):** If you want to edit the alert before it is sent you have to create a post filter.

The block diagram that comprises an alert is as follows:

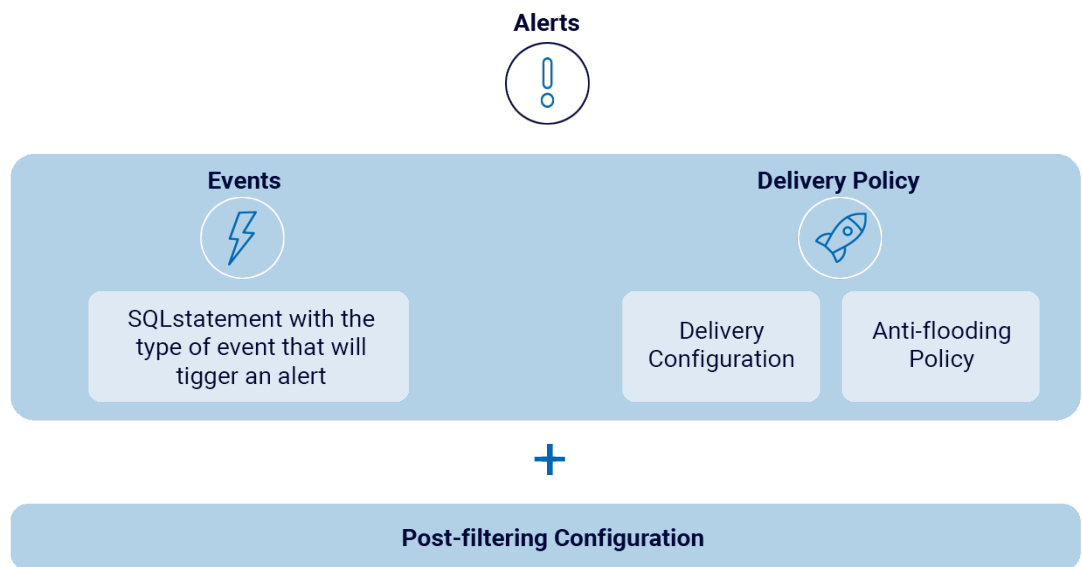


Figure 6.2: logical components that form an alert

Creating alerts

Alerts are created from the associated knowledge table. To create an alert, follow these steps.

1. Select the corresponding table in the **Search** side menu.

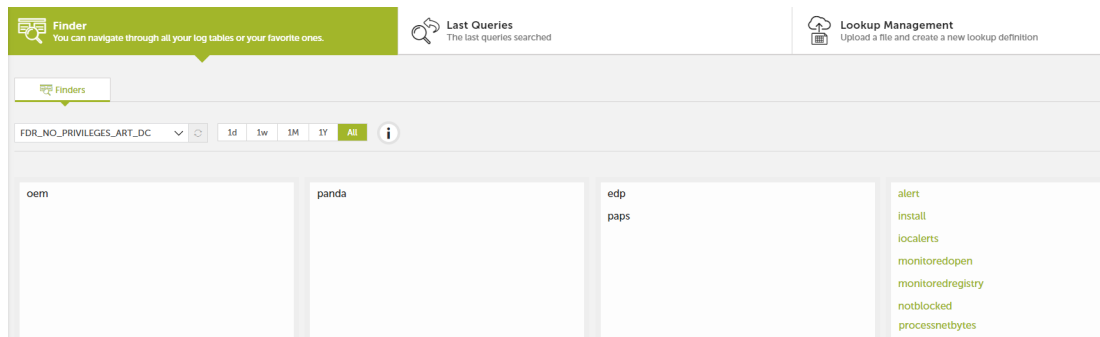



Figure 6.3: Alert management window

2. Apply the filters and data transformations required to generate the information you want and click the  icon in the toolbar.
3. Set the alert parameters.
 - **Message:** The alert subject.
 - **Description:** The alert content.
 - **Subcategory:** Tag that classifies the alert and enables later searches or filters.
 - **Alert Name:** New section.
4. Alert generation frequency.
 - **Each:** Generate an alert for each event entry in the table.
 - **Severall:** Generates a single alert for each group of events occurred during the specified period.
 - **Period:** Time interval.
 - **Threshold:** Number of events received in the specified period.
 - **Low:** Generates a single alert if fewer events occur than indicated for the specified period.
 - **Period:** Time interval.
 - **Threshold:** Number of events received in the specified period.
 - **Rolling:** Checks, at regular intervals, the events occurred during the specified period.
 - **Run every:** Indicates, in minutes, the frequency of checking for events.
 - **Check last:** Indicates the check interval.

If, for example, a **Period** of 5 minutes is set and a **Threshold** of 30, no alert will be sent until there are 30 events. Event 60 will generate a second warning and so on until the five-minute period has

concluded, at which time the event counter is reset to 0.



During the process of creating alerts, the volume of alerts generated according to the settings is checked. If the alert will generate more than 60 alerts per minute, the alert settings are invalid. In this case, increase the Threshold field to lower the number of alerts generated per minute.

Once the alert is created, the system will begin generating entries as the events defined in the alert occur. To view the generated alerts log, see the “[Alert management](#)” section later.

Alert management

The generated alerts can be managed by clicking the **Alerts** side menu. Click the **Alerts panel** tab to display the following sections: **Alerts Overview** and **Alerts History**.

Alerts Overview

This view displays the alerts generated by the system through various charts.

To access the alerts, click **Alerts**  from the side menu and then click the **Alerts Dashboard** tab from the top menu.



Figure 6.4: toolbar to configure alert lists

- **Type of chart (1):** This lets you choose the way that the alerts will be represented:
 - Line chart.
 - Timeline.
 - Calendar chart.
 - Voronoi diagram.
- **Enable/disable pie chart (2)**
- **Time period represented in the chart (3).**
 - 1 hour.
 - 6 hours.
 - 12 hours.
 - 1 day.
 - 1 week.
 - 1 month.


- 1 year
- **Filter by alert status** (4)
 - **Open:** Only open alerts are displayed.
 - **All alerts:** All alerts are displayed.



See “Tables and charts” on page 27 for more details about each type of chart

Alerts History

This section shows a list of the alerts generated. Each alert has a number of fields that the system fills in as configured by the administrator when creating the alert:


- **Status:** Watched; not read.
- **Type:** Type of alert, taken from the Message field in the alert settings, described in the section on “[Creating alerts](#)” earlier in the chapter.
- **Detailed Information:** Extract from the alert text taken from the Description field, described in the section on “[Creating alerts](#)” earlier in the chapter. Click the **Detailed Information** in the alert to display the content.
- **Category:** Alert category taken from the Subcategory and Context fields, described in the section on “[Creating alerts](#)” earlier in the chapter.
- **Priority:** All alerts are generated with normal priority by default. To change the priority of an alert (very low, low, normal, high, very high) you have to configure a postfilter. Refer to the point on “[Post filter management](#)” later in this guide.
- **Created:** Date and time of creation and the time elapsed since the alert was generated.
- **Menu:** The final column in the Alerts History table displays a menu with options for each alert:
- **View alerts details:** This lets you see all the information associated with the alert in a new window.
- **Create annotation:** This lets you add a text to the alert. Completing the form will add an  icon to the alert indicating that a technician made a comment about the alert. You can also convert a note into a task if the alert requires action over a period of time.
- **New filter:** This lets you create postfilters as described in the following section.
- **Mark as closed:** This lets you mark an alert as closed.
- **Delete**

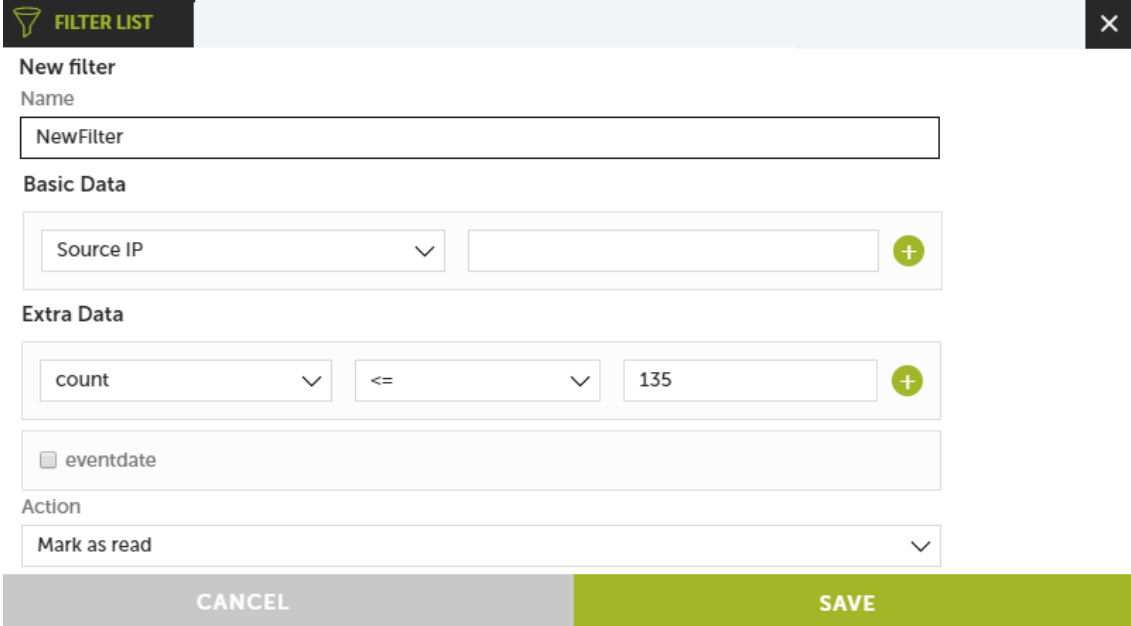
Establishing filters in the alerts history

Click the **Type**, **Category** or **Priority** fields of a specific alert to set a filter that will only display alerts that match the criteria set. The applied filters will be shown in the filter bar.

Creating post filters

Post filters allow you to edit the features of the generated alerts before they are sent, as well as deleting them if they coincide with certain criteria.

The post filters are created from the **Alerts** section in the side menu. Click the  icon of an alert that has been generated to display a drop-down menu with actions available. Click **New filter** to access the filter creation window



NEW FILTER

Name

NewFilter

Basic Data

Source IP

Extra Data

count <= 135

eventdate

Action

Mark as read

CANCEL SAVE

Figure 6.5: Window for creating a new post filter

The post filter screen comprises five sections:

Section 1: Description

This section specifies the name and criteria that alerts have to match for the filter to apply.

- **Name:** Name of the filter.
- **Context:** This sets the context of the alert as a filter condition.
- **Category:** This sets the category of the alert as a filter condition.
- **Priority:** This sets the priority of the alert as a filter condition.

Section 2: Basic data

This section is not used.

Section 3: Extra data

In this section you can set criteria based on the content which alerts must meet for the post filter to be applied.

In the process of configuring an alert, a series of columns can be established in the Counter field. The contents of these columns is accessible from the alert body when it is generated using the \$ symbol. The Extra data section allows you to choose from the dropdown menu those counters that you want to include as a filter condition.

Section 4: Filter dates

You can set one or more date ranges to act as a criteria. The post filter will not apply to alerts generated outside the established period.

Section 5: Action

- Mark as read.
- Change priority.
- False positive.
- Change notify method.
- Delete.

Click the **Save** button when you have finished setting all parameters. The filter code will be displayed in a pop-up window. Click **Add** to add it to the post filters window.

Post filter management

You can manage post filters from the **Alerts** side menu, by clicking **Post filters**.

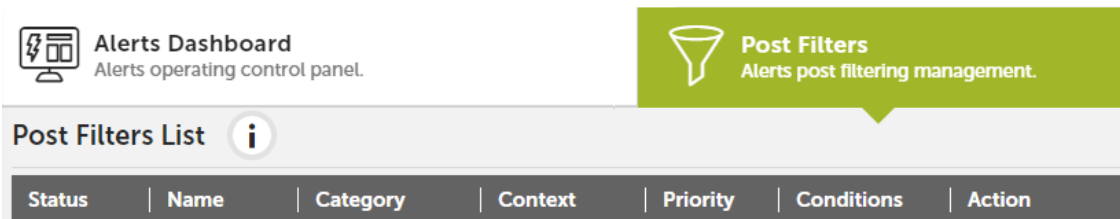


Figure 6.6: Post filter management tab

This window displays a list of the post filters configured with the following information:

- **Status:** Enabled or disabled.
- **Name:** Name given to the post filter when it was created.
- **Category:** Category that determines whether the post filter is applied.
- **Context:** Context that determines whether the post filter is applied.
- **Priority:** Alert priority that determines whether the post filter is applied.
- **Conditions:** Alert content that determines whether the post filter is applied.
- **Action:** Internal command that the alert will apply.

Creating delivery conditions

The delivery conditions are created through the side menu **Administration, Alerts Configuration**, then select the tab **Delivery methods**.

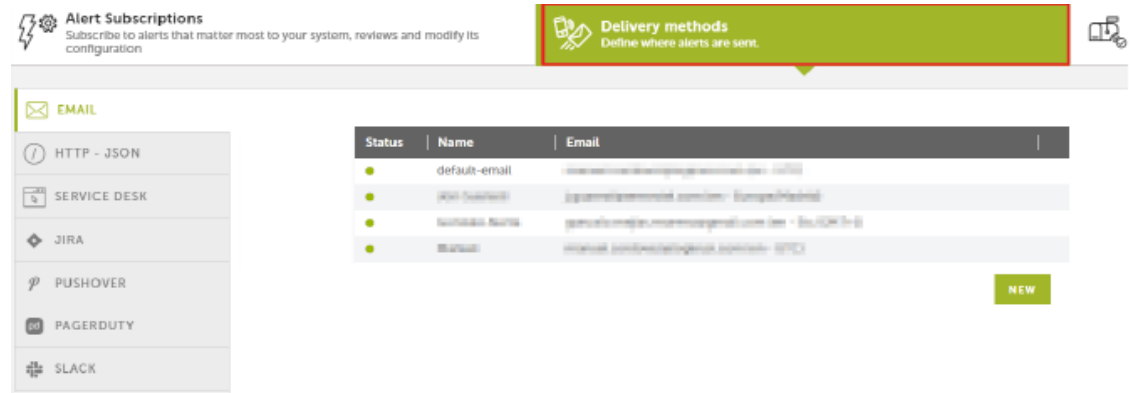


Figure 6.7: Creating delivery conditions

Select the delivery type in the left panel. The options are as follows:

- **Email:** The alerts are sent via email.
- **HTTP-JSON:** The alerts are sent via JSON objects.
- **Service desk:** The alerts are sent via Service Desk.
- **JIRA:** The alerts are sent via Jira server.
- **Pushover:** The alerts are sent in a Pushover account.
- **Pagerduty:** The alerts are sent in a PagerDuty account.
- **Slack:** The alerts are sent via the Slack service.

Once the type of delivery is selected, click the **New** button to set up a new type of delivery.

Email

This enables the sending of real-time alerts to email accounts.

The required fields are:

- **Name:** Name of the delivery method.
- **Email:** Email account of the recipient.
- **Timezone:** Sets the time and date for sending the email.
- **Language:** The language in which the alert is received.

HTTP-JSON

This enables the sending of real-time alerts via HTTP or HTTPS using JSON objects with POST method.

To improve security, in addition to using the HTTPS encryption protocol you can also enable Digest authentication.

The required fields are:

- **Name:** Name of the delivery method.
- **URL:** URL of the target server, specifying the protocol (http or https) and the port (e.g. <http://localhost:8080/index.php>).
- **Timezone:** Sets the time and date for sending the email.
- **Language:** The language in which the alert is received.
- **User:** This is only used when the **Authenticated** checkbox is selected.
- **Password:** This is only used when the **Authenticated** checkbox is selected.

Once the settings have been saved, an HTTP message is sent with a code to validate the server. In the list of JSON Delivery methods, the new configuration will be displayed preceded by a red dot (status, pending validation). By clicking the red dot, a window will open requesting the code sent to the server. Once the delivery settings are entered, it will be fully operational.

Service desk

This enables the real-time sending of alerts to Service Desk Plus servers, using two different methods: REST and SERVLET.

The required fields are:

- **Name:** Name of the delivery settings.
- **URL:** URL of the target server.
- **REST:** [http://\[SERVER\]:\[PORT\]/sdpapi/request/](http://[SERVER]:[PORT]/sdpapi/request/)
- **SERVLET:** [http://\[SERVER\]:\[PORT\]/servlets/RequestServlet](http://[SERVER]:[PORT]/servlets/RequestServlet)
- **Delivery method:** REST or SERVLET
- **User:** Name of the technician assigned.
- **Technician Key:** Technician key generated in the Service Desk administration panel.
- **Timezone:** Sets the time and date for sending the message.
- **Language:** The language in which the alert is received.

Once the settings have been saved, an HTTP message is sent with a code to validate the server. In the list of Service Desk delivery methods, the new configuration will be displayed preceded by a red dot (status, pending validation). By clicking the red dot, a window will open requesting the code sent to the server. Once the delivery settings are entered, it will be fully operational.

JIRA

This enables the real-time sending of alerts to Jira servers.

The required fields are:

- **Name:** Name of the delivery settings.
- **URL:** URL of the target server (e.g. <http://localhost:8090/rest/api/2/issue>).
- **User:** JIRA user name.
- **Password:** JIRA password.
- **Issue Type:** The type of task to be created in Jira. In the server URL, there will be a Json object with the projects created. The variable *issuetypes* will list the types of incidents permitted by the project.
- **Project key:** Identifier of the project where the alert will be created. In the server URL, there will be a Json object with the projects created and their identifiers. The Key tag contains the identifiers of each project.
- **Timezone:** Sets the time and date for sending the message.
- **Language:** The language in which the alert is received.

Once the settings have been saved, an HTTP message is sent with a code to validate the server. In the list of JIRA delivery methods, the new configuration will be displayed preceded by a red dot (status, pending validation). By clicking the red dot, a window will open requesting the code sent to the server. Once the delivery settings are entered, it will be fully operational.

Pushover

This enables the real-time sending of alerts to PushOver servers.

The required fields are:

- **Name:** Name of the delivery method.
- **Token Application:** API Key of the application created in <https://pushover.net/apps>
- **User/group:** API Key of the user or group to whom the alerts will be sent.
- **Device (optional):** Name of the device to which the alerts will be sent.
- **Title (optional):** Text that appears in the alert.
- **URL (optional):** Link sent in all alerts.
- **Url Title (optional):** Text that links to the URL above.
- **Sound (optional):** Type of notification to be sent.
- **Timezone:** Sets the time and date for sending the message.
- **Language:** The language in which the alert is received.

Once the settings have been saved, an HTTP message is sent with a code to validate the server. In the list of PushOver delivery methods, the new configuration will be displayed preceded by a red dot (status, pending validation). By clicking the red dot, a window will open requesting the code sent to the server. Once the delivery settings are entered, it will be fully operational.

Pagerduty

This enables the real-time sending of alerts to PagerDuty accounts.

The required fields are:

- **Name:** Name of the delivery method.
- **Service Key:** API Key of the PagerDuty service that receives the alert.
- **Client:** Name or identifier that appears in the alert.
- **Client URL U (optional):** Link sent in all alerts.
- **Timezone:** Sets the time and date for sending the message.
- **Language:** The language in which the alert is received.

Once the settings have been saved, an HTTP message is sent with a code to validate the server. In the list of PagerDuty delivery methods, the new configuration will be displayed preceded by a red dot (status, pending validation). By clicking the red dot, a window will open requesting the code sent to the server. Once the delivery settings are entered, it will be fully operational.

SLACK

This enables the real-time sending of alerts via SLACK.

The required fields are:

- **Name:** Name of the delivery settings.
- **Timezone:** Lets you set the time and date for sending the alert.
- **Channel:** Channel through which the alert is received.
- **Webhook URL:** URL of the target server.
- **Language:** Language in which the alert is received.

Once the settings have been saved, an HTTP message will be sent with a code to validate the server. Also, in the list of **Slack** delivery settings, the new settings will be displayed preceded by a red dot (status, pending validation). Click the red dot to open a window prompting you to enter the code sent to the server. Once entered, the delivery settings will be fully functional.

Delivery method management

Each of the Delivery methods created has a menu that allows it to be edited and/o deleted.

When editing a delivery method already created, a window is displayed with editing options.

Creating antiflooding policies

An antiflooding policy allows complete, temporary suspension of alert generation when the rate of alerts exceeds a certain threshold defined by the administrator in the policies.

Antiflooding policy creation is done from the side menu **Administration, Alerts Configuration**, then go to the **Alert Policies** tab, then the **Antiflooding Policy** tab.

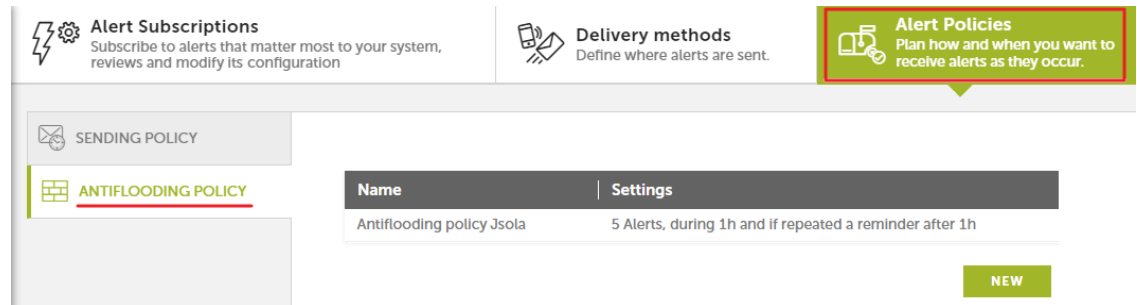


Figure 6.8: Creating antiflooding policies

Click **New** to display a window with the complete settings options of the policy.

Here you can set:

- Maximum number of alerts that can be received.
- Time period to which the previous criteria applies.
- A reminder if the alert is repeated after the established time period.

Editing antiflooding policies

Each of the antiflooding policies created has an associated menu that allows it to be edited and/or deleted.

When editing antiflooding policies already created, a window is displayed with editing options.

Creating alert policies or delivery methods

Alert policies, also called sending policies, let you define how the alerts generated are sent.

A sending policy is the nexus of the policies defined above (antiflooding policy and delivery methods).

Creating sending policies is carried out through the side menu **Administration, Alerts Configuration**, then go to the **Alert Policies** tab, then the **Sending Policy** tab.

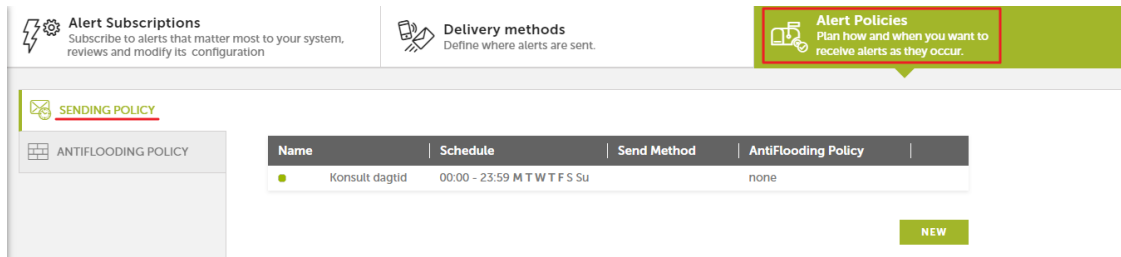


Figure 6.9: Creating alert policies or delivery methods

Click **New** to display a window with the complete settings options of the sending policy:

- **Name:** Name of the sending policy.
- **Default:** This indicates whether the policy is to be treated as a default policy. If there are alerts that don't have a sending policy assigned, this will be assigned by default.
- **Antiflooding policy:** This specifies the antiflooding policy to apply.
- **Schedule:** This indicates the time period when the policy will be active.
- **Send method:** This indicates the methods of delivery configured earlier that will be used to deliver the alert.

Editing sending policies

Each of the sending policies created has an associated menu that allows it to be edited and/or deleted.

When editing sending policies already created, a window is displayed with editing options.

Configuring an alert sending policy

Sending policies are assigned to alerts through the side menu Administration, Alerts Configuration, then go to the Alert Subscriptions tab.

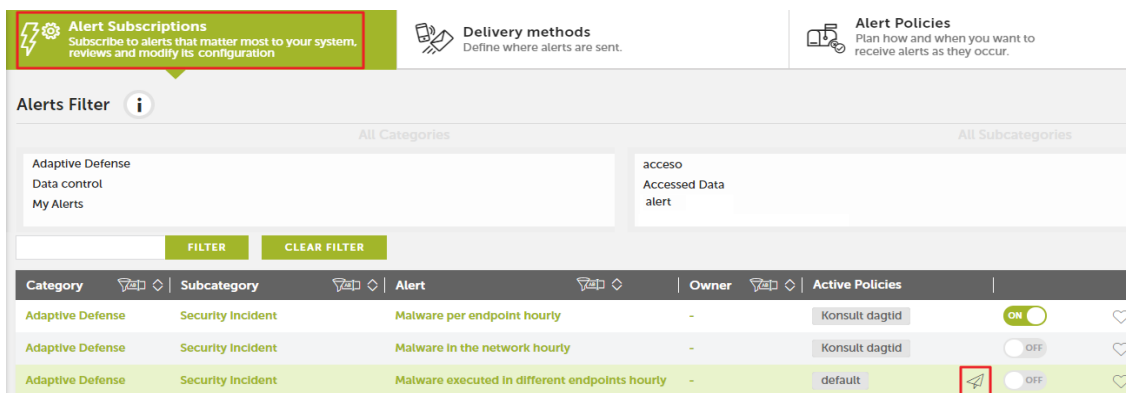



Figure 6.10: Configuring an alert sending policy

Each alert has an  icon which lets you select a sending policy.



Part 3

Additional information

Chapter 7: Knowledge table

Chapter 8: Hardware, software and network requirements

Chapter 7

Knowledge table

Panda Adaptive Defense sends all the information collected by Panda Adaptive Defense to the Advanced Reporting Tool service, which organizes it into easy-to-read tables.

Each line of a table is an event monitored by Panda Adaptive Defense. The tables contain a series of specific fields, as well as common fields that appear in all of them and which provide information such as when the event occurred, the computer where it was detected, its IP address, etc.

Terminology used in fields - - - - -	-73
Alert - - - - -	-74
10 most attacked and infected computers	76
10 most viewed threats	77
Other useful information	80
Install - - - - -	-80
Agent uninstall	81
Monitoredopen - - - - -	-81
Access to user documents	82
MonitoredRegistry - - - - -	-83
Notblocked - - - - -	-84
Ops - - - - -	-85
ProcessNetBytes - - - - -	-88
Graphical representation of the applications that use the most data	88
Registry - - - - -	-92
Persistence of installed threats	93
Socket - - - - -	-93
Programs that most connect to external computers	95
ToastBlocked - - - - -	-99
URLdownload - - - - -	100
Domains that receive most downloads requests	101
VulnerableAppsFound - - - - -	104
Computers with most vulnerable applications	105

Terminology used in fields

Many fields use prefixes that help refer to the information shown. The two most used prefixes are:

- **Parent:** The fields that begin with the Parent tag (parentPath, parentHash, parentCompany...) reflect the content of a characteristic or attribute of the parent process.

- **Child:** The fields that begin with the Child tag (childPath, childHash, childCompany...) reflect the content of a characteristic or attribute of a child process created by the parent process.

Besides these prefixes, many fields and values use abbreviations; knowing their meaning helps interpret the field in question:

- **Sig:** Digital signature
- **Exe:** Executable
- **Prev:** Prevalence
- **Mw:** Malware
- **Sec:** Seconds
- **Op:** Operation
- **Cat:** Category
- **PUP:** Potentially Unwanted Program
- **Ver:** Version
- **SP:** Service Pack
- **Cfg:** Configuration
- **Svc:** Service
- **PE:** Executable program
- **Cmp** and **comp:** Compressed file
- **Dst:** Destination

Listed below are the available tables indicating the type of information they contain and their specific fields.

Alert

This table reflects the incidents displayed in the **Activity** panel of the Panda Adaptive Defense dashboard.

It contains a line for each threat detected on the customer's network with information on the computer involved, type of incident, timestamp and result.

Name	Explanation	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
machineIP	IP address of the customer's computer that triggered the alert.	IP address

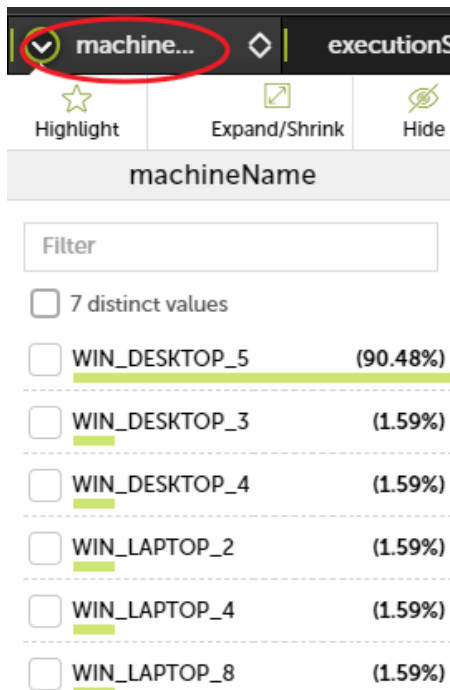
Table 7.1: Description of the alert type events generated

Name	Explanation	Values
date	Date of the user's computer when the event was generated.	Date
alertType	Category of the threat that triggered the alert.	<ul style="list-style-type: none"> Malware PUP
machineName	Name of the customer's computer.	String
executionStatus	The threat was run or not.	<ul style="list-style-type: none"> Executed Not Executed
dwelTimeSecs	Time in seconds from the first time the threat was seen on the customer's network.	Seconds
itemHash	Hash of the detected threat.	String
itemName	Name of the detected threat.	String
itemPath	Full path of the file that contains the threat.	String
sourceIP	If the malware came from outside the customer's network, this indicates the IP of the remote computer	IP address
sourceMachineName	If the malware came from outside the customer's network, this indicates the name of the remote computer.	String
sourceUserName	If the malware came from outside the customer's network, this indicates the user of the remote computer.	String
urlList	List of accessed URLs if a browser exploit is detected.	String
docList	List of accessed documents if a file exploit is detected.	String
version	Content of the Version attribute of the process metadata.	String
vulnerable	Indicates if the application is considered vulnerable or not.	Boolean

Table 7.1: Description of the alert type events generated

Since the **Alerts** table is a transposition of the **Activity** panel in the **Panda Adaptive Defense** console, it is easy to obtain statistics of the most affected computers:

10 most attacked and infected computers



Click the header of the **machineName** or **machineIP** columns to obtain a list of the 10 most attacked computers.

Figure 7.1: Drop down with the most frequently found values

This list covers from the time when Panda Adaptive Defense first started to work on the customer's network; if you want to reduce the range, you can simply narrow down the interval with the **Apply interval** controls.



Figure 7.2: Tool to define the data to display

These lists include both malware blocking and executions; if you want to only show infected computers, you will need to add a filter by clicking the icon in the toolbar.



Figure 7.3: Access to the tool to create a new data filter

You will also need to configure a data filter using the **executionStatus** field and equaling it to **Executed**, as shown in the image.

The screenshot shows the 'OPERATIONS OVER COLUMNS' interface with the 'FILTER DATA' section active. Under 'Operation', 'normal' is selected. Under 'Case sensitivity', 'insensitive' is selected. The 'Arguments' section shows 'executionStatus' in the 'Value' field and 'executed' in the 'is equal (ignoring case) to' field. At the bottom, there are 'CANCEL' and 'FILTER DATA' buttons.

Figure 7.4: Tool to create a new data filter

10 most viewed threats

Similarly, by clicking the **itemHash** or **itemName** columns you can display quick statistics on the 10 most viewed threats on the customer's network.

Another way of obtaining far more visual information is to generate a chart of the most viewed malware. The name of the malware is shown on the coordinate axis and the number of occurrences on the abscissa axis.

For this, you need to follow the steps below:

1. Add an aggregation to the **itemName** field without any time limit (**No temporal aggregation**).



Figure 7.5: Access to the tool to create a new data pool

OPERATIONS OVER COLUMNS

FILTER DATA

Operation
 normal negated
 Equal - case insensitive (eqic) ▼ ⓘ

Case sensitivity
 sensitive insensitive all

Arguments NEW ARGUMENT

Value ▼ ✎ 🗑️

is equal (ignoring case) to ▼ ✎ 🗑️

CANCEL **FILTER DATA**

Figure 7.6: Tool to create a data pool

2. Add a counter function to determine how many occurrences there are in each **itemName** group.

OPERATIONS OVER COLUMNS ✕

CREATE COLUMN FILTER DATA GROUP BY **AGGREGATE FUNCTION** OR

Column Name

Aggregation ▼ ⓘ

Arguments NEW ARGUMENT

No arguments

CANCEL **AGGREGATE FUNCTION**

Figure 7.7: Tool to create operations on columns

3. Add a filter to differentiate the aggrupation of 2 or fewer occurrences. This will clean the chart of

those threats that have only been viewed twice.

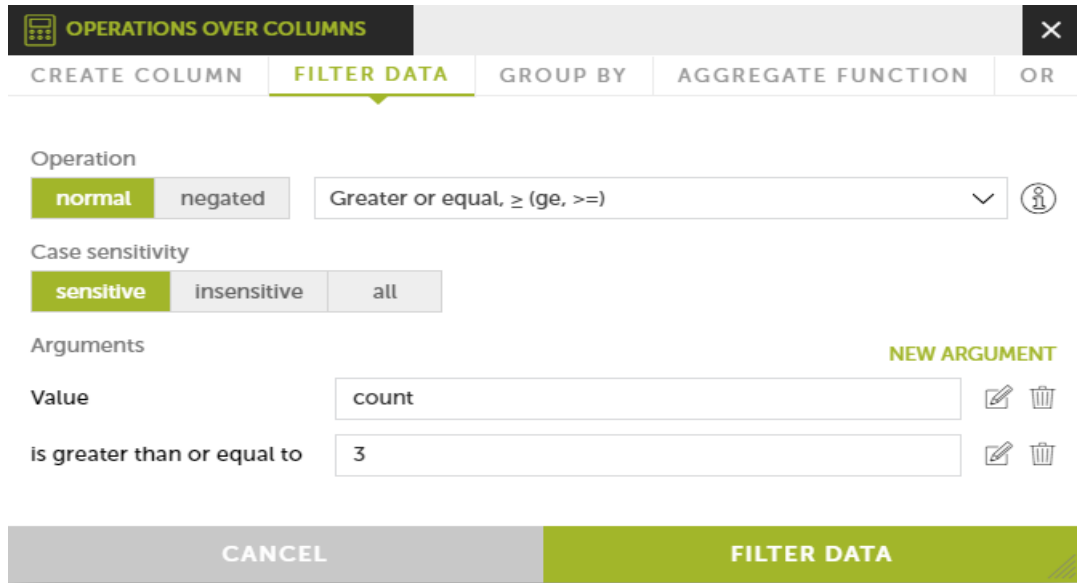


Figure 7.8: Tool to create operations on columns

4. Add a **Chart Aggregation** type chart and use the **Count** column as a parameter.

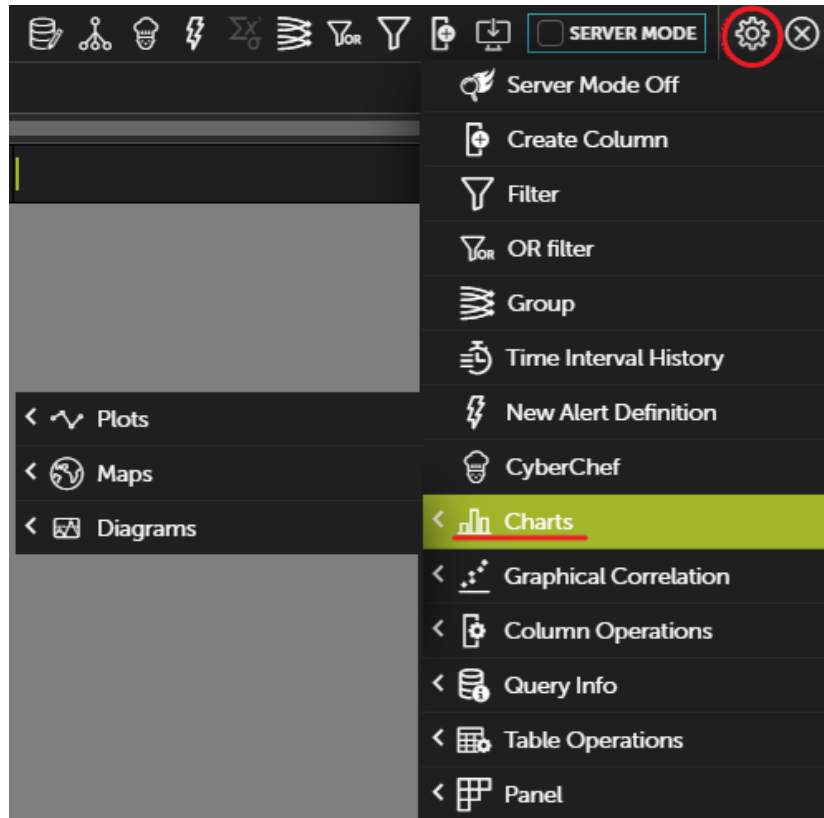


Figure 7.9: Graphical access menu

At this point, you'll have a list of incidents grouped by threat, with the number of occurrences for each threat. You can create a simple chart with this data:

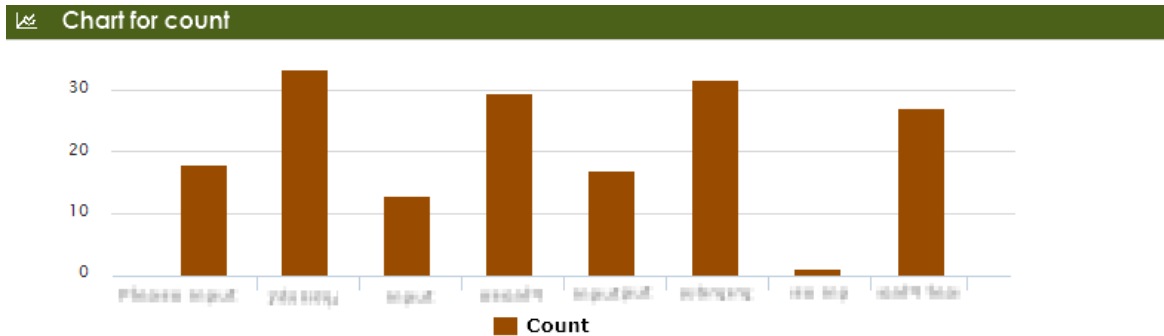


Figure 7.10: Generated bar graph

Other useful information

There are several interesting fields in the **Alerts** table that can be used to extract valuable information on the attacks received on the customer's network:

- **Eventdate:** Grouping by this field you can see the number of daily attacks and determine if there is an ongoing epidemic.
- **dwellTimeSecs:** This field provides the detection window of the threats received, i.e. the time from when the threat was first seen on the customer's network to its classification.
- **itemHash:** Given that the name of the threat varies among security vendors, the hash field can be used to group threats instead of the **itemName**. This also helps to distinguish malware that is labeled with the same name.

Install

This table logs all the information generated during the installation of the Panda Adaptive Defense agents on the customer's computers.

Name	Explanation	Values
eventDate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
serverdate	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
machineIP1	IP address of an additional network card if installed.	IP address
machineIP2	IP address of an additional network card if installed.	IP address

Table 7.2: Description of the events generated in the installation process

Name	Explanation	Values
op	Operation performed.	<ul style="list-style-type: none"> • Install • Uninstall • Upgrade
osVersion	Operating system version.	String
osServicePack	Service Pack version.	String
osPlatform	Platform of the operating system installed: <ul style="list-style-type: none"> • Darwin_x86_64: macOS (64-bit) • Win64NT: Windows (64-bit) • Win32NT: Windows (32-bit) • Linux_i686: Linux (32-bit) • Linux_x86_64: Linux (32-bit) • Win64ARM: Windows for ARM processors 	Enumeration

Table 7.2: Description of the events generated in the installation process

Agent uninstall

To find those computers whose agent has been uninstalled in a specific time interval, set the date range and add a filter on the **op** field in order to select all the rows that have the “Uninstall” string. This way, you will get a list of all the computers whose agent has been uninstalled and are therefore vulnerable to threats.

Monitoredopen

This table logs the data files accessed by the applications run on the user's computer, and the processes that accessed user data.

Name	Explanation	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
serverdate	Date of the user's computer when the event was generated.	Date
date	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	Process user name.	String

Table 7.3: Description of the events generated by the monitored processes when accessing data files

Name	Explanation	Values
muid	Internal ID of the customer's computer.	String in the following format xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx
parentHash	Digest/hash of the file that accessed data.	String
parentPath	Path of the process that accessed data.	String
parentValidSig	Digitally signed process that accessed data.	Boolean
parentCompany	Content of the Company attribute of the metadata of the file that accesses data.	String
parentCat	Category of the file that accessed data.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
parentMWName	Malware name if the file that accessed data is classified as a threat.	<ul style="list-style-type: none"> • String • Null if the item is not malware
childPath	Name of the data file accessed by the process. By default, only the file extension is indicated to preserve the privacy of the customer's data.	String
loggedUser	User logged in on the computer at the time of file access.	String
firstParentCat	Initial classification of the parent file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null

Table 7.3: Description of the events generated by the monitored processes when accessing data files

Access to user documents

As this table shows the files accessed by all processes run on the user's computer, it is quite simple to locate an information leak in case of infection.

Filter by the **parentCat** field to distinguish goodware from other possibilities. This way, you will obtain a list of accesses to data files by unclassified processes or processes classified as malware, which will allow you to see at a glance the impact of data leakage and take the necessary measures.

MonitoredRegistry

This table logs every attempt to modify the registry as well as registry accesses related to permissions, passwords, certificate stores and other.

Name	Explanation	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
date	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	User name of the process that accessed or modified the registry.	String
muid	Internal ID of the customer's computer.	String in the following format xxxxxxx-xxxx-xxxx- xxxx-xxxxxxxxxxxx
parentHash	Digest/hash of the process that accessed or modified the registry.	String
parentPath	Path of the executable that accessed or modified the registry.	String
parentValidSig	Digitally signed process that accessed the registry.	Boolean
parentCompany	Content of the Company attribute of the metadata of the process that accessed the registry.	String
parentCat	Process category.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
parentMwName	Malware name if the process is classified as a threat.	String Null if the item is not malware
regAction	Operation performed on the computer registry.	<ul style="list-style-type: none"> • CreateKey • CreateValue • ModifyValue
key	Affected registry branch or key.	String
value	Name of the affected value under the registry key.	String
valueData	Value content.	String

Table 7.4: Description of the events generated by the processes monitored when accessing the registry

Name	Explanation	Values
loggedUser	User logged in on the computer at the time of registry access.	String
firstParentCat	Initial classification of the parent file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null

Table 7.4: Description of the events generated by the processes monitored when accessing the registry

Notblocked

This table logs the items that Panda Adaptive Defense has not scanned due to exceptional situations such as service timeout on startup, configuration changes, etc.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
date	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	Process user name.	String
muid	Internal ID of the customer's computer.	String in the following format xxxxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxxx
parentHash	Digest/hash of the parent file.	String
parentPath	Parent process path.	String
parentValidSig	Digitally signed parent process.	Boolean
parentCompany	Content of the Company attribute of the parent process metadata.	String
parentCat	Parent file category.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring

Table 7.5: Description of the events generated by the non-monitored elements

Name	Description	Values
ParentmwName	Malware name if the parent file is classified as a threat.	<ul style="list-style-type: none"> String Null if the item is not malware
childHash	Child file digest/hash.	String
childPath	Child process path.	String
childValidSig	Digitally signed child process.	Boolean
childCompany	Content of the company attribute of the child process metadata.	String
childCat	Child process category.	<ul style="list-style-type: none"> Goodware Malware PUP Unknown Monitoring
childMWName	Malware name if the child file is classified as a threat.	<ul style="list-style-type: none"> String Null if the item is not malware
firstParentCat	Initial classification of the parent file that performed the logged operation.	<ul style="list-style-type: none"> Goodware Malware PUP Unknown Monitoring Null
firstChildCat	Initial classification of the child file that performed the logged operation.	<ul style="list-style-type: none"> Goodware Malware PUP Unknown Monitoring Null

Table 7.5: Description of the events generated by the non-monitored elements

Ops

This table logs all operations performed by the processes seen on the customer's network.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date

Table 7.6: Description of the events generated by the monitored processes

Name	Description	Values
serverdate	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	Process user name.	String
op	Operation performed.	<ul style="list-style-type: none"> • CreateDir • Exec • CreatePE • DeletePE • LoadLib • OpenCmp • RenamePE • CreateCmp
muid	Computer's unique ID.	String in the following format xxxxxxxxxxxx-xxxx-xxxx- xxxxxxxxxxxx
parentHash	Parent file digest/hash.	String
parentDriveType	Type of drive where the parent process resides.	<ul style="list-style-type: none"> • Fixed • Remote • Removable
parentPath	Parent process path.	String
parentValidSig	Digitally signed parent process.	Boolean
parentCompany	Content of the Company attribute of the parent file metadata.	String
parentCat	Parent file category.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
parentMWName	Name of the malware found in the parent file.	<ul style="list-style-type: none"> • String • Null if the item is not malware
childHash	Child file digest/hash.	String
childDriveType	Type of drive where the child process resides.	<ul style="list-style-type: none"> • Fixed • Remote • Removable
childPath	Child process path.	String
childValidSig	Digitally signed child process.	Boolean

Table 7.6: Description of the events generated by the monitored processes

Name	Description	Values
childCompany	Content of the Company attribute of the child file metadata.	String
childCat	Child file category.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
childMWName	Name of the malware found in the child file.	<ul style="list-style-type: none"> • String • Null if the item is not malware
Ocs_Exec	Whether software considered as vulnerable was run or not.	Boolean
Ocs_Name	Name of the software considered vulnerable.	String
OcsVer	Version of the software considered vulnerable.	String
action	Action performed.	<ul style="list-style-type: none"> • Allow • Block • BlockTimeout
serviceLevel	Agent mode: <ul style="list-style-type: none"> • Learning: The agent allows the execution of unknown processes. • Hardening: The agent prevents the execution of processes classified as threats. • Block: The agent prevents the execution of processes classified as threats and unknown processes. 	Enumeration
params	Command-line execution parameters of the process run.	Character string
firstParenCat	Initial classification of the parent file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null

Table 7.6: Description of the events generated by the monitored processes

Name	Description	Values
firstChildCat	Initial classification of the child file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null

Table 7.6: Description of the events generated by the monitored processes

ProcessNetBytes

This table logs the data usage of the processes seen on the customer's network. A log per process is sent approximately every four hours with the amount of data transferred since the last log was sent. The total amount of bytes sent and received per process will be the sum of all quantities received.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
serverdate	Date of the user's computer when the event was generated.	Date
machineName	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
version	Version of the Panda Adaptive Defense agent.	String
user	Process user name.	String
muid	Internal ID of the customer's computer.	String in the following format xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
hash	Digest/hash of the process.	String
path	Program name and path.	String
bytesSent	Number of bytes sent by the process since the last event was generated.	Numeric
bytesReceived	Number of bytes received by the process since the last event was generated.	Numeric

Table 7.7: Description of events generated by network traffic generated processes

Graphical representation of the applications that use the most data

This table is most typically used to see which programs on the network computers use the most data. It is worth noting that this table doesn't differentiate between internal data and external data usage. That is, the total amount of data used by a process may be a mixture of data requested over the

Internet and data obtained from the company's internal servers (mail servers, Intranet Web servers, files shared among workstations, etc.).

To be able to easily determine which network applications use the most data, a Voronoi diagram will be generated with the data received by each application run on the customer's network.

1. Extract the name of the program run

As the name of each application run is logged in the Path field with its full path, the first step will be to extract the application name. To do that, create a new column named **PreProgramName** with the **Substitute All** operation and the following parameters:

- **String to scan:** Path column
- **Regular expression:** (.*\)\)
- **Template:** (empty)

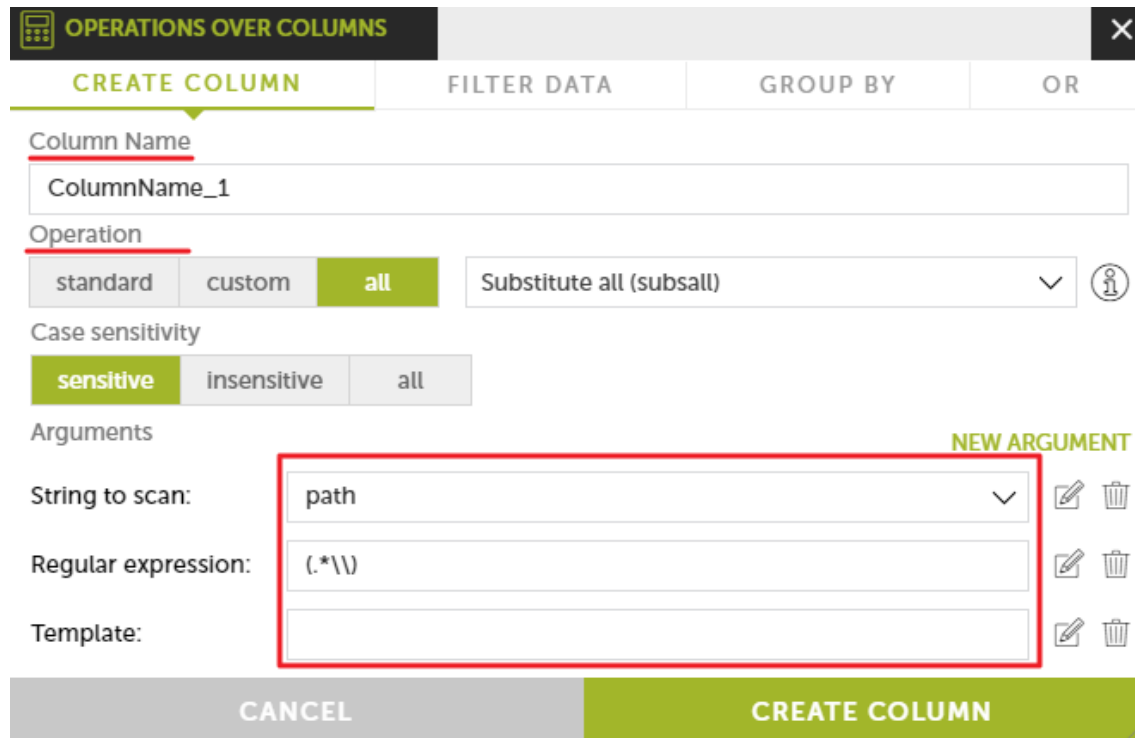


Figure 7.11: Tool to create a new column

Then, filter by **null** to avoid processing wrong entries, and create another column **-ProgramName-** with the **Lower Case** operation over the previously created column (**ProgramName**). This way, you'll get the names of all programs run in lowercase letters and without errors.

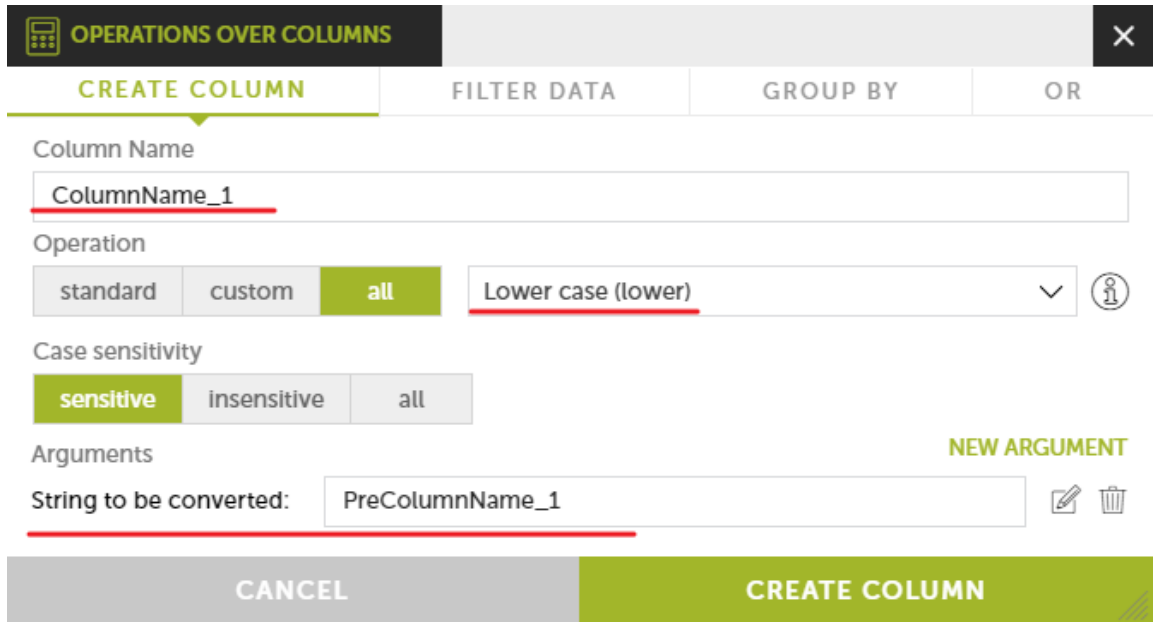


Figure 7.12: Tool to create a new column

Another simpler method would be to use the table's hash field to identify running processes. This method, however, may result in a higher number of unique processes as each version of a program has its own hash value, which would make reading the diagram generated in the last step more difficult.

2. Add a daily aggregation

Add an aggregation based on the number of days to cover (a daily aggregation in our example) along with the **ProgramName** field.

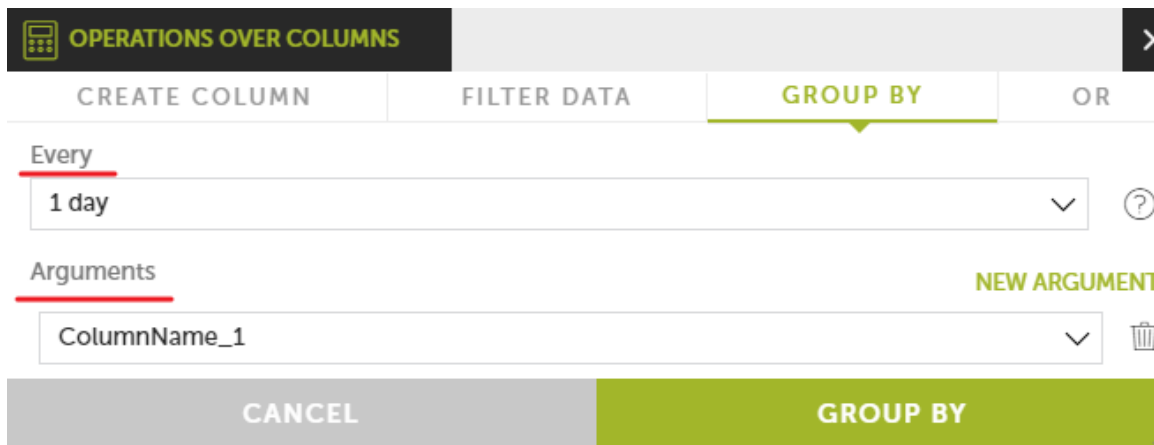


Figure 7.13: Tool to create a new grouping

3. Add a sum function

Add a sum function over the **bytesReceived** field to sum the total number of bytes received by each process.

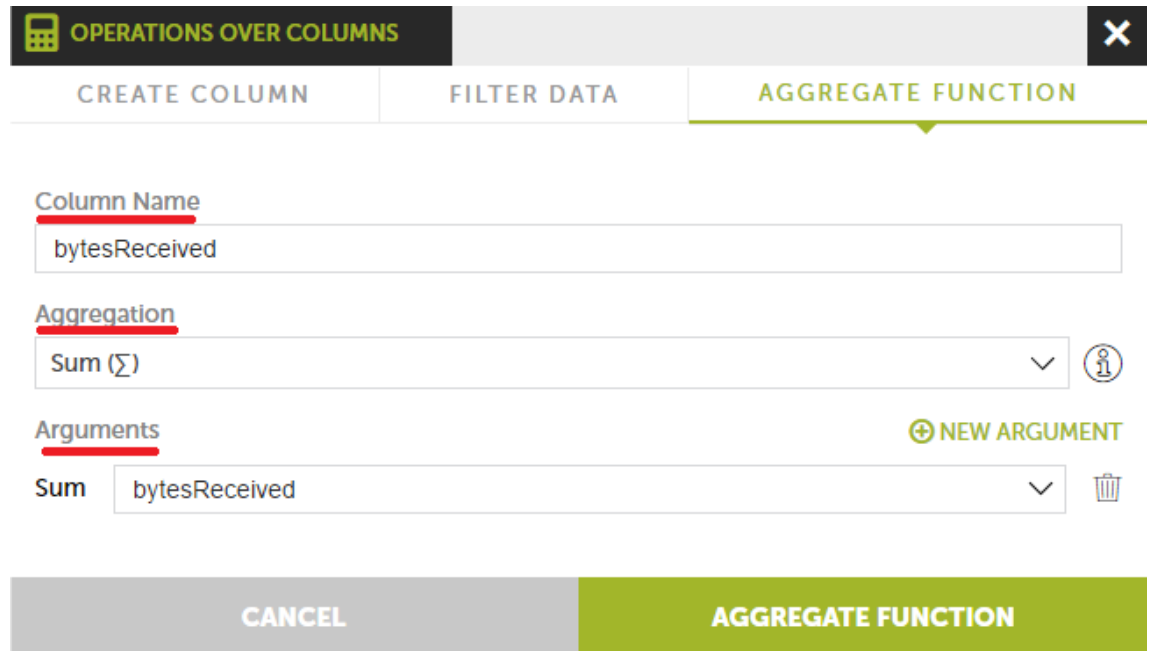


Figure 7.14: Tool to create operations on columns

4. Add a data filter

In order to see only the processes that have used more than a certain amount of data and simplify the diagram, you can filter the results by a figure: for example, 100 megabytes (104857600 bytes).

5. Create the Voronoi diagram

Drag the **ProgramName** field to the **Signals** section. Then, drag the **bytesReceived** field to the **Value** section.

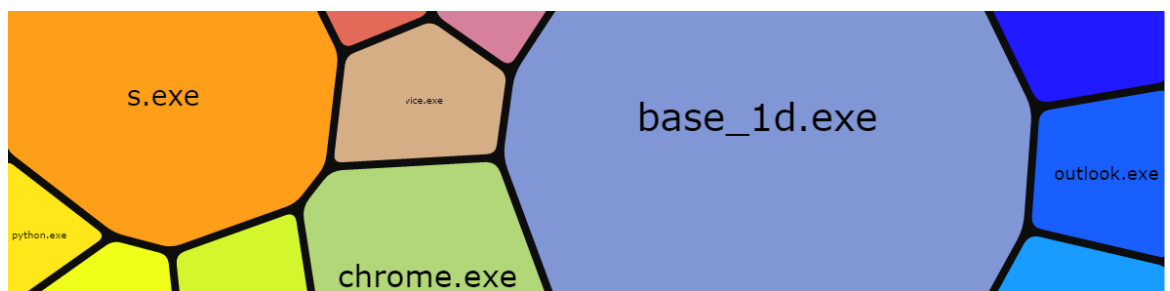


Figure 7.15: Resulting Voronoi chart

Registry

This table logs all operations performed on the registry branches used by malicious programs to become persistent and survive computer restarts.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Toolserver.	Date
serverdate	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	User name of the process that modified the registry.	String
op	Operation performed on the computer registry.	<ul style="list-style-type: none"> • ModifyExeKey • CreateExeKey
hash	Digest/hash of the process that modified the registry.	String
muid	Computer's unique ID.	String in the following format xxxxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxxx
targetPath	Path of the executable that the registry key points to.	Type of drive where the process that accessed the registry resides
regKey	Registry key.	String
driveType	Type of drive where the process that accessed the registry resides.	String
path	Path of the process that modified the registry.	String
validSig	Registry key.	Boolean
company	Registry key.	String
Cat	Process category.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
mwName	Malware name if the process is classified as a threat.	<ul style="list-style-type: none"> • String • Null if the item is not malware

Table 7.8: Description of the events generated by the processes that modify the branches of the registry to gain persistence

Name	Description	Values
firstCat	Category of the process the first time it was classified.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring

Table 7.8: Description of the events generated by the processes that modify the branches of the registry to gain persistence

Persistence of installed threats

This table logs all accesses to the registry by the processes run on the user's computer when they affect those branches that are read when the system starts up as part of the operating system boot process. These branches are modified by malware to ensure it runs on every boot up.

There are many registry branches that allow a program to be run at startup, but the most used by Trojans and other types of threats are:

```

HKCU\Software\Microsoft\Windows\CurrentVersion\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\Run
HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run
HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnceEx
HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnceEx

```

Socket

This table logs all network connections established by the processes seen on the customer's network.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Toolserver.	Date
serverdate	Date of the user's computer when the event was generated.	Date
machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	Process user name.	String

Table 7.9: Description of the events generated by the processes that use the network

Name	Description	Values
hash	Digest/hash of the process that established the connection.	String
driveType	Type of drive where the process that established the connection resides.	<ul style="list-style-type: none"> • Fixed • Remote • Removable
path	Path of the process that established the connection.	String
protocol	Communications protocol used by the process.	<ul style="list-style-type: none"> • TCP • UDP • ICMP • ICMPv6 • IGMP • RF
remotePort	Destination port the process communicates with.	0-65535
direction	Communication direction.	<ul style="list-style-type: none"> • Upload • Download • Bidirectional • Unknown
remoteIP	Destination IP address.	IP address
localPort	Source IP address.	0-65535
localIP	IP v6 destination address.	IP address
validSig	Digitally signed file that established the connection.	Boolean
company	Content of the Company attribute of the metadata of the file that established the connection.	String
category	Current category of the process that established the connection.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
mwName	Malware name if the process that established the connection is classified as a threat.	<ul style="list-style-type: none"> • String • Null if the item is not malware

Table 7.9: Description of the events generated by the processes that use the network

Name	Description	Values
firstCategory	Category of the process the first time it was classified.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
times	<p>Number of times the same communication event has occurred in the last hour. For two communication events to be considered the same, the following parameters plus the communication direction must be the same:</p> <ul style="list-style-type: none"> • The process name. • The local IP address of the process. • The process path. • The target IP address of the communication. • The target port of the communication. <p>The first time a communication is detected, an event is sent with the times field set to 1. Later, for each hour that passes after the first event, the times field will indicate the number of equal communication events that have occurred in that time span minus 1, along with the date of the last event logged.</p>	Numeric

Table 7.9: Description of the events generated by the processes that use the network

Programs that most connect to external computers

You can create a chart with the external computers that the legitimate software run on the network most connect to. For this, you need to follow the steps below:

1. Add a filter that removes all programs that are not considered legitimate. For this, you need to set the **Cat** field to "Goodware".
2. Add a filter that removes all connections to private IP addresses. For this, you need to create a

column with **the Is Public IPv4** operation on the **dstip** field, as shown in the figure:

The screenshot shows the 'OPERATIONS OVER COLUMNS' tool interface. The 'CREATE COLUMN' tab is active. The configuration is as follows:

- Column Name:** usrcol1
- Operation:** Is Public IPv4 (selected from a dropdown menu)
- Case sensitivity:** sensitive (selected from a group of buttons)
- Arguments:** IP to test: dstip

Buttons for 'CANCEL' and 'CREATE COLUMN' are visible at the bottom.

Figure 7.16: Tool to add a data filter

3. Add both **latitude** and **longitude** columns that extract the longitude and latitude from the **dstIP** field with the operations **Geolocated Latitude/Longitude**.

The screenshot shows the 'OPERATIONS OVER COLUMNS' tool interface. The 'CREATE COLUMN' tab is active. The configuration is as follows:

- Column Name:** lat
- Operation:** Geolocated Latitude (mmlatitude) (selected from a dropdown menu)
- Case sensitivity:** sensitive (selected from a group of buttons)
- Arguments:** Ip: 000.000.000.000

Buttons for 'CANCEL' and 'CREATE COLUMN' are visible at the bottom.

Figure 7.17: Tool to create a column

At this point, you'll have a list of connections from legitimate software to public IP addresses, and the latitude and longitude of each IP address. The coordinates obtained will be shown on the map-type

chart as dots.

As the intention is to show the number of connections to the same IP address, you will need to form an aggregation and add a counter to obtain the number of IP addresses repeated in the aggregation.

4. Add an aggregation with the arguments **dstIP**, **latitude** and **longitude**, without time limit (**No temporal aggregation**).

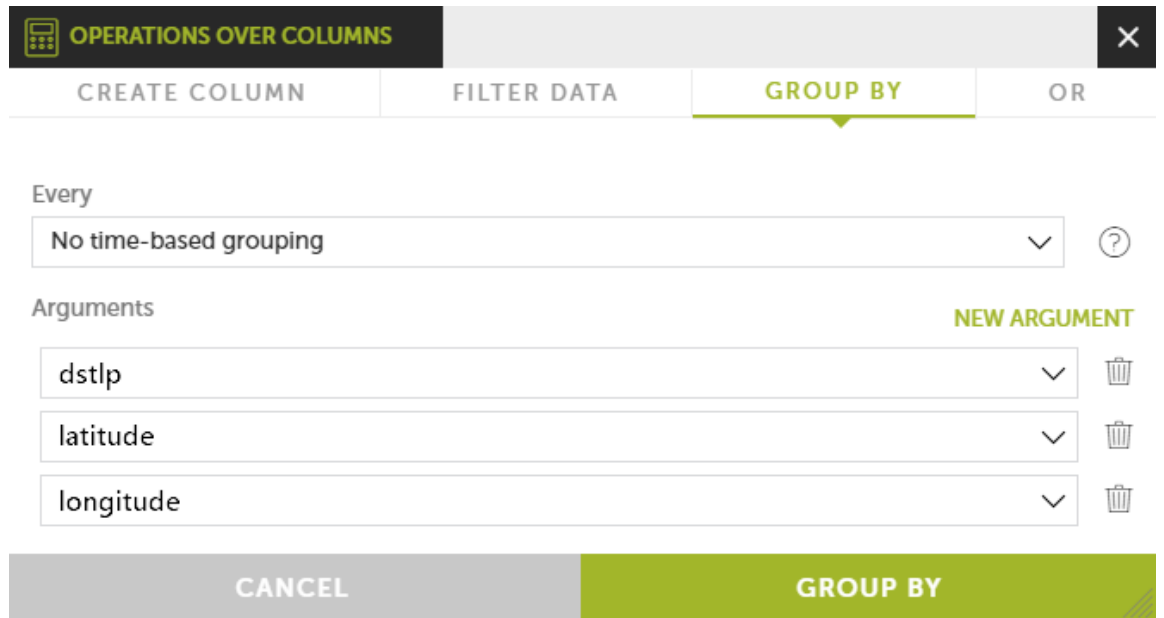


Figure 7.18: tool to create a data pool

5. Add a counter-type function.
6. Add a **Flat world map by coordinates** or **Google heat map** chart using the **count**, **latitude** and **longitude** columns as data.

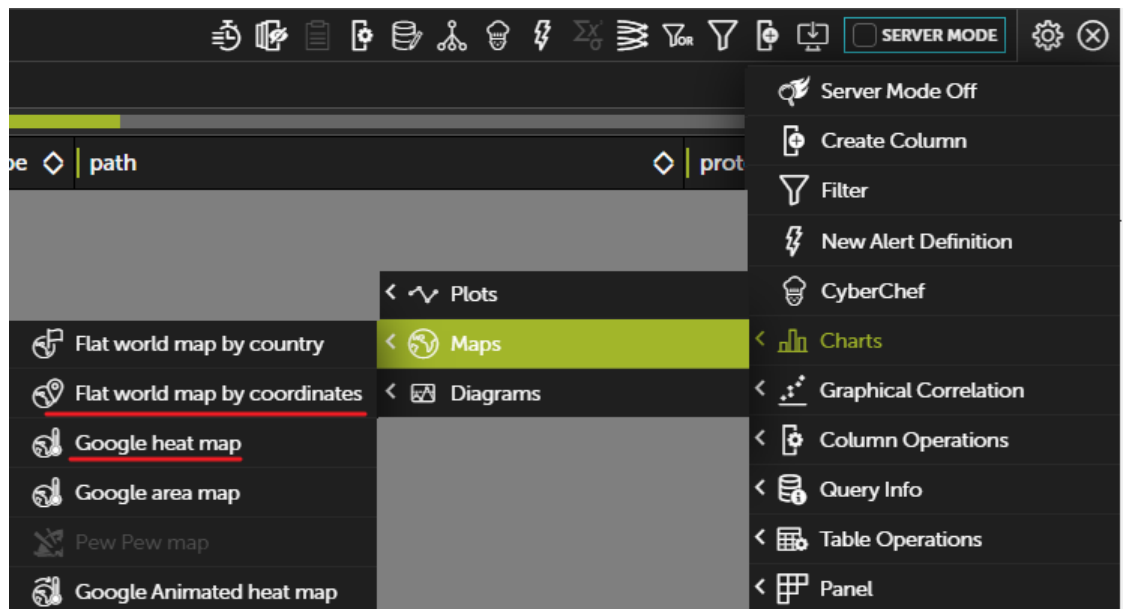


Figure 7.19: Access to chart creation

When dragging the columns to the relevant boxes, the map will show the relevant data with dots in different colors and sizes.

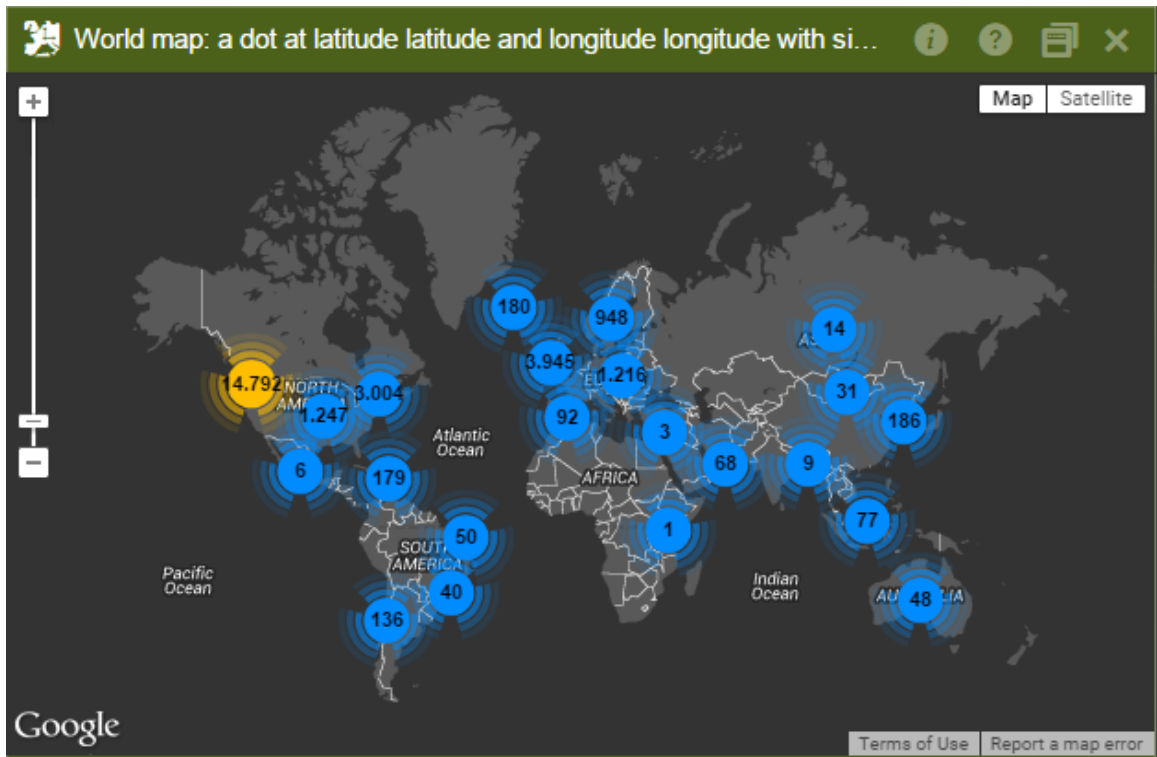


Figure 7.20: resulting world map graphic

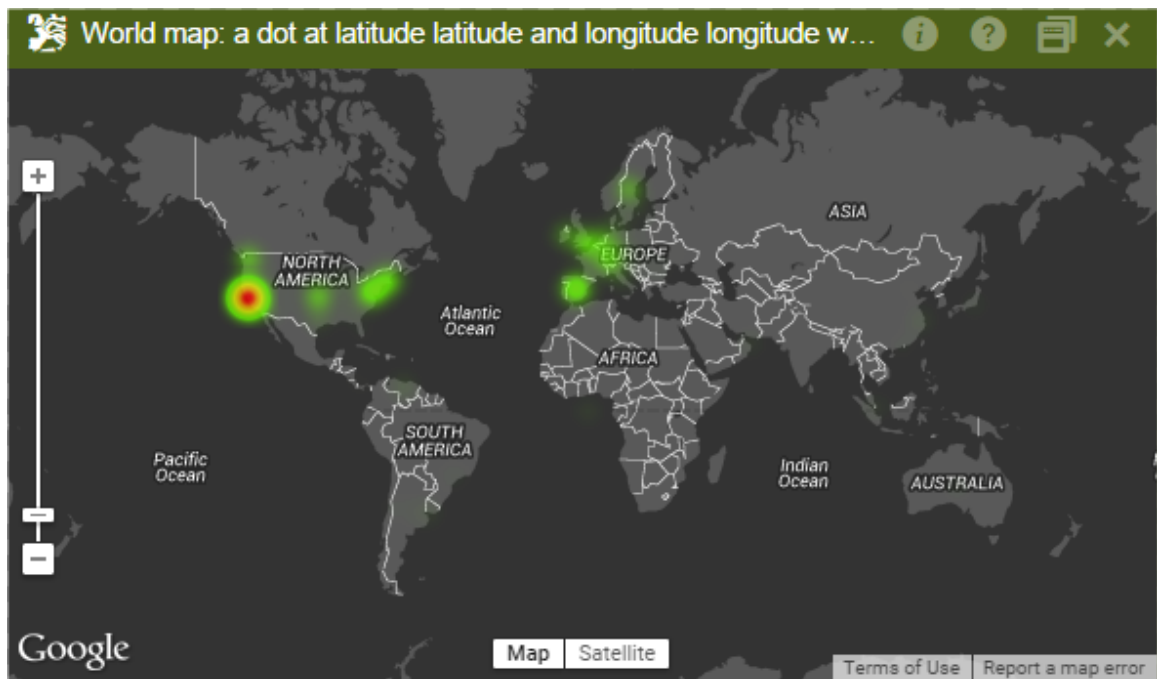


Figure 7.21: resulting world map graphic

ToastBlocked

This table contains a record for each blocked process, as Panda Adaptive Defense has not yet returned the relevant classification.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
serverdate	Date of the user's computer when the event was generated.	Date
machineName	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
user	User name of the process blocked.	String
muid	Computer's unique ID.	String in the following format xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
hash	Digest/hash of the process blocked.	String
path	Path of the process blocked.	String
toastBlockReason	<ul style="list-style-type: none"> • 0 OK: The customer accepts the message. • 1 Timeout: The pop-up message disappears due to non-action by the user. • 2 Angry: The user rejects the block action. • 3 Block • 4 Allow • 5 BadCall 	Enumerator
toastResult	Result of the pop-up message: <ul style="list-style-type: none"> • 0 OK: The customer accepts the message. • 1 Timeout: The pop-up message disappears due to non-action by the user. • 2 Angry: The user rejects the block action. • 3 Block • 4 Allow • 5 BadCall 	Enumerator

Table 7.10: Description of blocking events because the process executed is unknown

URLdownload

This table contains information on the HTTP downloads performed by the processes seen on the customer's network (URL, downloaded file data, computers that downloaded data, etc.)

Name	Explanation	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Tool server.	Date
serverdate	Date of the user's computer when the event was generated.	Date
Machine	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
User	Process user name.	String
muid	Internal ID of the customer's computer.	String in the following format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
url	Download URL.	URI stem
parentHash	Digest/hash of the process that downloaded the file.	String
parentDriveType	Type of drive where the process that downloaded the file resides.	<ul style="list-style-type: none"> • Fixed • Remote • Removable
parentPath	Path of the process that downloaded the file.	String
parentValidSig	Digitally signed process that downloaded the file.	Boolean
parentCompany	Content of the Company attribute of the metadata of the process that downloaded the file.	String
parentCat	Category of the process that downloaded the file.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
parentMwname	Malware name if the process that downloaded the file is classified as a threat.	<ul style="list-style-type: none"> • String • Null if the item is not malware
childHash	Digest/hash of the downloaded file.	String
childDriveType	Type of drive where the process that downloaded the file resides.	<ul style="list-style-type: none"> • Fixed • Remote • Removable
childPath	Path of the downloaded file.	String

Table 7.11: Description of the files downloaded by the processes via HTTP

Name	Explanation	Values
childValidSig	Digitally signed downloaded file.	Boolean
childCompany	Content of the company attribute of the downloaded file metadata.	String
childCat	Category of the downloaded file.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring
childMwname	Malware name if the downloaded file is classified as a threat.	<ul style="list-style-type: none"> • String • Null if the item is not malware
firstParentCat	Initial classification of the parent file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null
firstChildCat	Initial classification of the child file that performed the logged operation.	<ul style="list-style-type: none"> • Goodware • Malware • PUP • Unknown • Monitoring • Null

Table 7.11: Description of the files downloaded by the processes via HTTP

This table displays all downloads performed by users on the network regardless of whether they are malware or goodware. In addition to filtering by download information, you can also view a chart showing the domains that receive most download requests.

Domains that receive most downloads requests

To show this type of information, you need to manipulate the content of the **URL** field to remove the part of the string not of interest to you and end up with the domain.

1. Create a new column with the **Split** operation on the **URL** field

The screenshot shows the 'OPERATIONS OVER COLUMNS' interface with the 'CREATE COLUMN' tab selected. The configuration is as follows:

- Column Name:** Domain
- Operation:** Split (selected from a dropdown menu)
- Case sensitivity:** sensitive (selected from buttons: sensitive, insensitive, all)
- Arguments:**
 - Split: url
 - by separator: /
 - and return piece: 2

Buttons at the bottom include 'CANCEL' and 'CREATE COLUMN'.

Figure 7.22: tool for creating a column

2. Group by different **URL** selecting **No temporal aggregation**

The screenshot shows the 'OPERATIONS OVER COLUMNS' interface with the 'GROUP BY' tab selected. The configuration is as follows:

- Every:** No time-based grouping (selected from a dropdown menu)
- Arguments:** domain

Buttons at the bottom include 'CANCEL' and 'GROUP BY'.

Figure 7.23: tool for creating a data pool

3. Add a counter-type aggregation column.

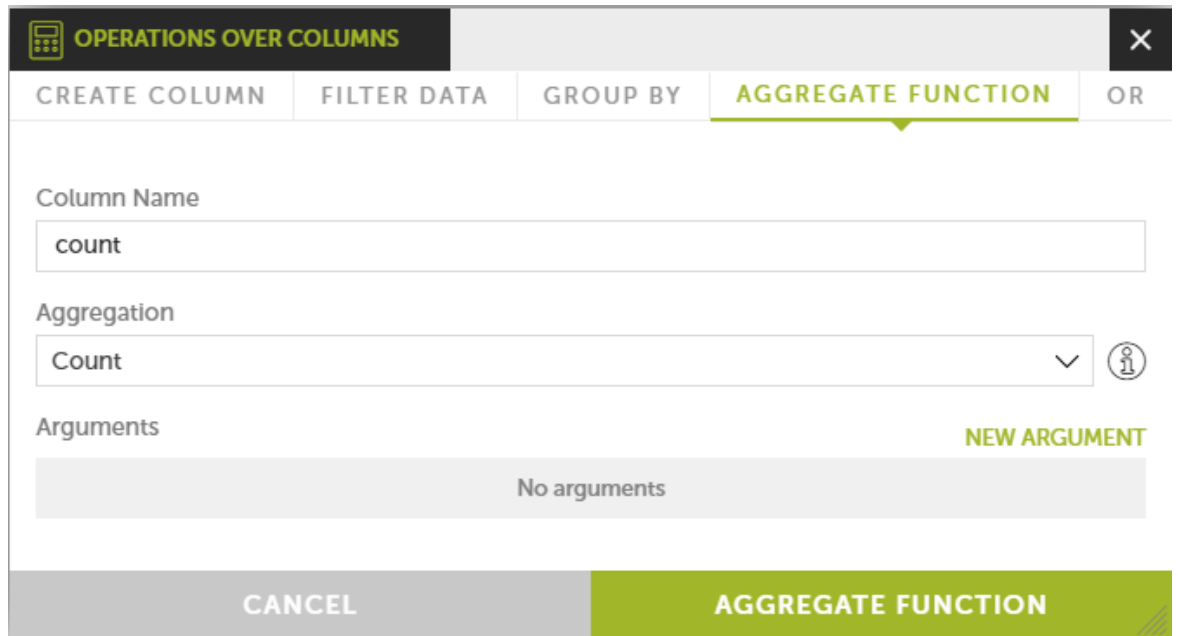


Figure 7.24: tool to add an operation on columns

This way, you will obtain a list for each grouped domain and the number of occurrences of each domain within each group. With this information, you can easily generate a chart with the most visited domains for downloading purposes.

In this case, a pie chart will be generated. Filter out the groups with fewer occurrences to be able to look in more detail at the rest of domains.

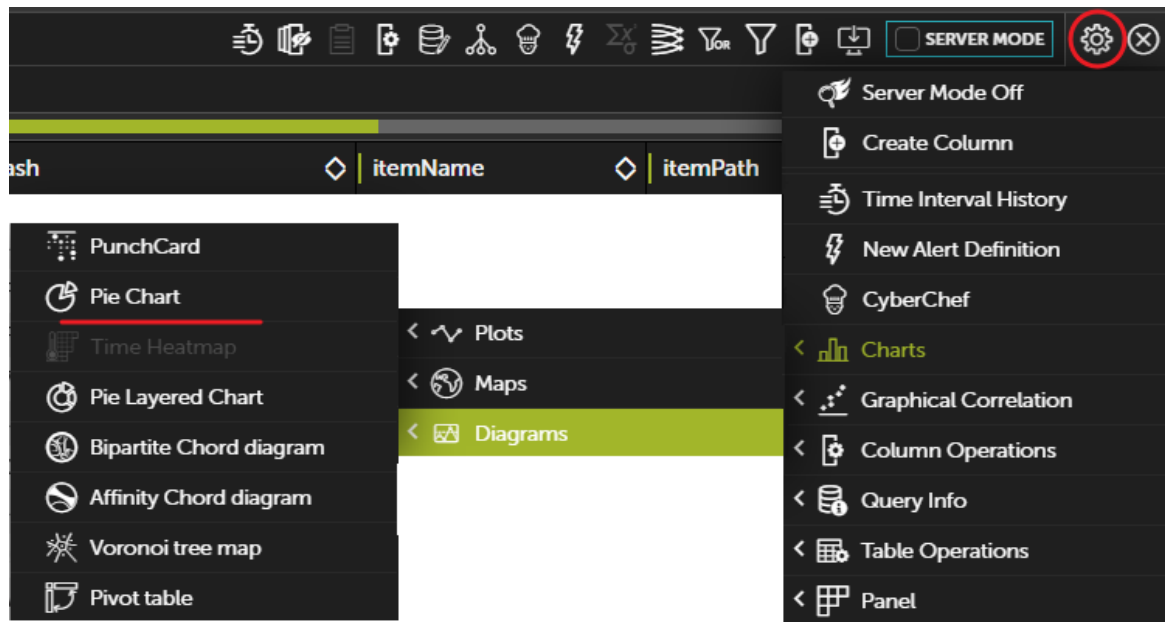


Figure 7.25: Access to chart creation

In pie charts, the different sections are active so when you pass the mouse over them they show the percentages and name of the items represented.

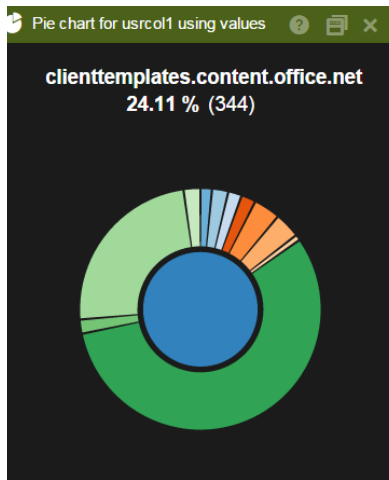


Figure 7.26: Resulting graph

• **Other useful information**

Similarly, other fields can be used and combined to enhance or filter the lists and obtain more refined tables. You can use the following fields:

- **Machine or machineIP:** Grouping by these fields you can see the computers on the customer's network that start the most downloads.
- **ParentCat and ChildCat:** Filtering by these fields you can clear the table and only show what is classified as malware. You can therefore obtain the domains considered as malware domains in order to block them using a Layer-7 firewall.

VulnerableAppsFound

This table logs every vulnerable application found on each computer on the customer's network. Unlike the **Ops** table, whose **ocsExec**, **ocsName** and **ocsVer** fields show the vulnerable applications that have been run on the network, this table shows all of the vulnerable applications that reside on computers. Once every day, a log is sent per each detected application. If an application is deleted, the solution will stop sending the relevant event.

Name	Description	Values
eventdate	Date when the event was received on the Panda Advanced Reporting Toolserver.	Date
serverdate	Date of the user's computer when the event was generated.	Date
muid	Internal ID of the customer's computer.	String in the following format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
machineName	Name of the customer's computer.	String
machineIP	IP address of the customer's computer.	IP address
criticalSoftEventType	Indicates the existence of vulnerable software.	Present
itemHash	Digest of the vulnerable program found on the computer.	String
fileName	Name of the vulnerable file.	String

Table 7.12: Description of vulnerable applications found

Name	Description	Values
filePath	Full path of the vulnerable file.	String
internalName	Content of the Name attribute of the vulnerable file metadata.	String
companyName	Content of the Company attribute of the vulnerable file metadata.	String
fileVersion	Content of the Version attribute of the vulnerable file metadata.	String
productVersion	Content of the ProductVersion attribute of the vulnerable file metadata.	String

Table 7.12: Description of vulnerable applications found

Computers with most vulnerable applications

This table is typically used to determine which computers on the network have most vulnerable applications.

In this example, no distinction is made between installed applications and applications that have simply been copied to the computer's hard disk. Also, bear in mind that an application copied N times to a computer doesn't count as one, but as N.

1. Add a 1-day aggregation

As vulnerable software events are generated on a daily basis, you can select to group all rows every day with the **machineName field as argument**. However, bear in mind that those computers that have not connected to the server on a particular day won't generate any events.

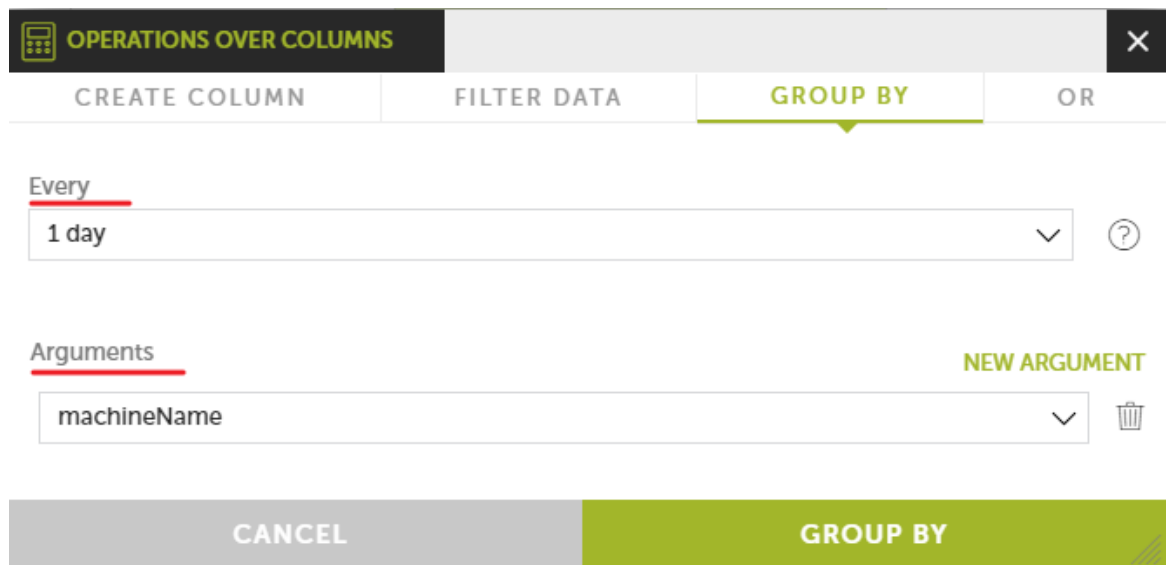


Figure 7.27: Tool for creating a new grouping

2. Add a **Count** function.

As each vulnerable program found on a computer generates one event per day, it will be enough to count the number of times that each computer appears in the aggregation.

3. Add a filter.

If the values obtained are too dispersed, you may want to set a filter that excludes those computers that don't reach a certain threshold. To do that, simply add a **Greater or equal** filter with the appropriate value. Below that threshold there will be no computers on the list.

4. Generate a Voronoi diagram

Use the **MachineName** field as **Signal** and the **Count** field as **Value** to generate a diagram that shows the most vulnerable computers on the network.

Chapter 8

Hardware, software and network requirements

Panda Advanced Reporting Tool is a cloud service and, as such, the entire infrastructure required to provide the service to its customers is hosted on Panda Security's premises. This frees organizations from the need to deploy additional hardware or software across their corporate networks. Nevertheless, the computers and the network to protect need to meet a series of minimum requirements to ensure that the product works properly.

CHAPTER CONTENT

Management console access requirements	-----	107
Hardware requirements	-----	107

Management console access requirements

In order for you to access the Web console, your system must meet the following requirements:

- Have a certified/supported browser (others may be compatible)
 - Mozilla Firefox
 - Google Chrome



Other browsers may also work, but some of their versions may not be supported. That's why we recommend that the aforementioned Web browsers be used.

- Internet connection and communication through port 443.
- Minimum screen resolution 1280x1024 (1920x1080 recommended).

Hardware requirements

- Enough processing power to generate the module's charts and lists in real time.

- Enough bandwidth to display all the information collected from users' computers in real time.

