Agent labels

This feature provides the opportunity to customize alerts information from agents, giving us the chance of include specific information of each agent which could be useful when dealing with alerts. In addition, in large environments it could be used to distinguish groups of agents by any common characteristic like their time zone, for example.

The following sections shows how this feature works as well as an use case.

How it works

Configuring labels for being shown at alerts is too simple. It can be done using a simple XML structure which allows to add information with the format `key:value`, how to configure them it is explained at Labels section section of `ossec.conf`. It is remarkable that it exists the possibility of using dots to split “key” names nesting them in JSON formatted alerts.

It is also allowed to centralize it using `agent.conf`. This way, from the manager side it is possible to set labels for specific agents. Note that whether it exists a duplicated label key in `ossec.conf` and `agent.conf`, the second one will override the first one. For more information about usage of centralized configuration see its dedicated section: Centralized configuration.

In addition, more technical configuration is available in Internal configuration. Particularly, `analysisd.label_cache_maxage` and `analysisd.show_hidden_labels`.

Use case

It is interesting to think about a scenario where the use of labels could be useful. Let’s imagine we have a large environment deployed in Amazon Web Service (AWS) and monitored by Wazuh. In that situation, we want to know in the manager the following information about each agent when an alert is triggered:

- AWS instance-id.
- AWS Security group.
- Network IP address.
- Network MAC.
- Date of installation (hidden).

To include these labels into alerts of a specific agent, it is necessary to set the following configuration in `ossec.conf`:

```
<labels>
  <label key="aws.instance-id">i-052a1838c</label>
  <label key="aws.sec-group">sg-1103</label>
  <label key="network.ip">172.17.0.0</label>
  <label key="network.mac">02:42:ac:11:00:02</label>
  <label key="installation" hidden="yes">January 1st, 2017</label>
</labels>
```

Whether the configuration is set from the manager, the configuration has to be set in `agent.conf` using this format:

```
<agent_config name="92603de31548">
  <labels>
    <label key="aws.instance-id">i-052a1838c</label>
    <label key="aws.sec-group">sg-1103</label>
    <label key="network.ip">172.17.0.0</label>
    <label key="network.mac">02:42:ac:11:00:02</label>
    <label key="installation" hidden="yes">January 1st, 2017</label>
  </labels>
</agent_config>
```

When an alert is fired for this agent, the previous configuration will add to alerts the specified information:

```
** Alert 1488922301.778562: mail  - ossec,syscheck,pci_dss_11.5,
2017 Jun 07 13:31:43 (92603de31548) 192.168.66.1->syscheck
aws.instance-id: i-052a1838c
aws.sec-group: sg-1103
network.ip: 172.17.0.0
network.mac: 02:42:ac:11:00:02
Rule: 550 (level 7) -> 'Integrity checksum changed.'
Integrity checksum changed for: '/var/ossec/etc/ossec.conf'
Size changed from '3663' to '3664'
Old md5sum was: '98b351df146410f174a967d726f9965e'
New md5sum is : '7f4f5846dcaa0013a91bd6d3ac4a1915'
Old sha1sum was: 'c6368b866a835b15baf20976ae5ea7ea2788a30e'
New sha1sum is : 'c959321244bdcec824ff0a32cad6d4f1246f53e9'
```

And the same alert in JSON format shows the advantage of using splitting in “key” names:

```
{
  "timestamp": "2017-03-07T13:31:41-0800",
  "rule": {
    "level": 7,
    "description": "Integrity checksum changed.",
    "id": "550",
    "firedtimes": 1,
    "groups": [
      "ossec",
      "syscheck"
    ],
    "pci_dss": [
      "11.5"
    ]
  },
  "agent": {
    "id": "001",
    "name": "92603de31548",
    "ip": "192.168.66.1",
    "labels": {
      "aws": {
        "instance-id": "i-052a1838c",
        "sec-group": "sg-1103"
      },
      "network": {
        "ip": "172.17.0.0",
        "mac": "02:42:ac:11:00:02"
      }
    }
  },
  "manager": {
    "name": "ubuntu"
  },
  "full_log": "Integrity checksum changed for: '/var/ossec/etc/ossec.conf' Size changed from '3663' to '3664'
Old md5sum was: '98b351df146410f174a967d726f9965e' New md5sum is : '7f4f5846dcaa0013a91bd6d3ac4a1915' Old
sha1sum was: 'c6368b866a835b15baf20976ae5ea7ea2788a30e' New sha1sum is :
'c959321244bdcec824ff0a32cad6d4f1246f53e9'",
  "syscheck": {
    "path": "/var/ossec/etc/ossec.conf",
    "size_before": "3663",
    "size_after": "3664",
    "perm_after": "100640",
    "uid_after": "0",
    "gid_after": "999",
    "md5_before": "98b351df146410f174a967d726f9965e",
    "md5_after": "7f4f5846dcaa0013a91bd6d3ac4a1915",
    "sha1_before": "c6368b866a835b15baf20976ae5ea7ea2788a30e",
    "sha1_after": "c959321244bdcec824ff0a32cad6d4f1246f53e9",
    "event": "modified"
  },
  "decoder": {
    "name": "syscheck_integrity_changed"
  },
  "location": "syscheck"
}
```


Finally, when we have enabled email reports. It will be sent an email as follows:

```
Wazuh Notification.
2017 Mar 07 13:31:41

Received From: (92603de31548) 192.168.66.1->syscheck
Rule: 550 fired (level 7) -> "Integrity checksum changed."
Portion of the log(s):

aws.instance-id: i-052a1838c
aws.sec-group: sg-1103
network.ip: 172.17.0.0
network.mac: 02:42:ac:11:00:02
Integrity checksum changed for: '/var/ossec/etc/ossec.conf'
Old md5sum was: '98b351df146410f174a967d726f9965e'
New md5sum is : '7f4f5846dcaa0013a91bd6d3ac4a1915'
Old sha1sum was: 'c6368b866a835b15baf20976ae5ea7ea2788a30e'
New sha1sum is : 'c959321244bdcec824ff0a32cad6d4f1246f53e9'
```