Machinery Modülleri

Machinery modülleri, Cuckoo'nun sanallaştırma yazılımınızla nasıl etkileşimde bulunması gerektiğini tanımlar (veya potansiyel olarak fiziksel disk görüntüleme çözümleriyle bile). Belirli bir vendor zorlamama kararı aldığımızdan beri, sürüm 0.4'ten itibaren tercih ettiğiniz çözümü kullanabilir ve varsayılan olarak desteklenmiyorsa, Cuckoo'nun nasıl kullanılacağını tanımlayan özel bir Python modülü yazabilirsiniz.

Her makine modülü, `cuckoo/cuckoo/machinery/` dizini içinde bulunmalıdır, böylece `cuckoo.machinery` modülü altına düşer.

Temel bir machinary modülü şöyle görünecektir:

```
from cuckoo.common.abstracts import Machinery
from cuckoo.common.exceptions import CuckooMachineError

class MyMachinery(Machinery):
    def start(self, label):
        try:
            revert(label)
            start(label)
        except SomethingBadHappens:
            raise CuckooMachineError("oops!")

    def stop(self, label):
        try:
            stop(label)
        except SomethingBadHappens:
            raise CuckooMachineError("oops!")
```

Cuckoo için gereksinimler şunlardır:

- Class, `Machinery`'den alınmalıdır.
- `start()` ve `stop()` fonksiyonlarınız olmalıdır.
- Bir şey başarısız olduğunda `CuckooMachineError`'u yükseltmelisiniz.

Anlaşıldığı gibi, makine modülü bir Cuckoo kurulumunun temel bir parçasıdır, bu nedenle kodunuzu hata ayıklamak için yeterli zaman harcadığınızdan ve onu herhangi beklenmedik bir hataya karşı sağlam ve dirençli hale getirdiğinizden emin olun.

Konfigürasyon

Her makine modülü, `$CWD/conf/<makine modülü adı>.conf` (bu, Git deposundaki `cuckoo/data/conf/<makine>conf`'ye çevrilir) konumunda özel bir yapılandırma dosyasıyla birlikte gelmelidir. Örneğin, `cuckoo/cuckoo/machinery/kvm.py` için bir `$CWD/conf/kvm.conf`'ya sahibiz.

Yapılandırma dosyası varsayılan yapıyı izlemelidir:

```
[kvm]
# Specify a comma-separated list of available machines to be used. For each
# specified ID you have to define a dedicated section containing the details
# on the respective machine. (E.g. cuckoo1,cuckoo2,cuckoo3)
machines = cuckoo1

[cuckoo1]
# Specify the label name of the current machine as specified in your
# libvirt configuration.
label = cuckoo1

# Specify the operating system platform used by current machine
# [windows/darwin/linux].
platform = windows

# Specify the IP address of the current machine. Make sure that the IP address
# is valid and that the host machine is able to reach it. If not, the analysis
# will fail.
ip = 192.168.122.105
```

Bu konfigürasyon dosyasında, bir [<modül adı>] başlığı bulunmalı ve bir makine ID'lerini içeren bir `'machines'` alanına sahip olmalıdır.

Her bir makine için bir etiket, bir platform ve IP belirtmelisiniz.

Bu alanlar, Cuckoo'nun zaten gömülü `initialize()` fonksiyonunu kullanması için gereklidir. Bu fonksiyon, mevcut makinelerin listesini oluşturur.

Konfigürasyon yapısını değiştirmeyi planlıyorsanız, `initialize()` fonksiyonunu geçersiz kılmalısınız (kendi modülünüz içinde, Cuckoo'nun çekirdek kodunu değiştirmeniz gerekmez). Orijinal kodunu `cuckoo/common/abstracts.py` içindeki `Machinery` classta bulabilirsiniz.

LibVirt

Cuckoo 0.5 sürümüyle başlayarak, LibVirt üzerine yeni makine modülleri geliştirmek oldukça kolaydır. `cuckoo/common/abstracts.py` dosyasında zaten gerekli tüm işlevselliği sağlayan `LibVirtMachinery`'i bulabilirsiniz. Bu temel sınıftan miras alarak, aşağıdaki örnekte olduğu gibi bağlantı dizesini belirtmeniz yeterlidir:

Cuckoo 2.0.7a1 sürümüyle başlayarak, özel bir dsn kullanabilirsiniz. Bu dsn'yi kvm.conf dosyasında ayarlayarak kullanabilirsiniz. Örnek:

dsn = qemu+ssh://192.168.56.1/system

Bu, LibVirt tarafından desteklenen tüm sanallaştırma teknolojileri için çalışır. Sadece, LibVirt paketinizin (örneğin, Linux dağıtımınızdan kullanıyorsanız) ihtiyacınız olan teknolojiyi destekleyecek şekilde derlendiğinden emin olun.

Aşağıdaki komutla kontrol edebilirsiniz:

```
$ virsh -V
Virsh command line tool of libvirt 0.9.13
See web site at http://libvirt.org/

Compiled with support for:
 Hypervisors: QEmu/KVM LXC UML Xen OpenVZ VMWare Test
 Networking: Remote Daemon Network Bridging Interface Nwfilter VirtualPort
 Storage: Dir Disk Filesystem SCSI Multipath iSCSI LVM
 Miscellaneous: Nodedev AppArmor Secrets Debug Readline Modular
```

Eğer sanallaştırma teknolojiniz `Hypervisors` listesinde bulunmuyorsa, eksik olanı belirli destekle tekrar derlemeniz gerekecektir.

Revision #1
Created 19 January 2024 11:45:09 by Ertan Sözer
Updated 19 January 2024 11:46:40 by Ertan Sözer